IN THE UNITED STATES DISTRICT COURT
FOR THE EASTERN DISTRICT OF TEXAS
MARSHALL DIVISION

OPTi Inc.	§	
	§	**CIVIL ACTION 2:04-CV-377 (TJW)**
Plaintiff,	§	
	§	
v.	§	**JURY**
	§	
NVIDIA Corporation	§	
	§	
Defendant.	§	
	§	
	§	

CONTAINS CONFIDENTIAL AND "OUTSIDE COUNSEL EYES ONLY" INFORMATION OF THIRD PARTIES (INTEL, COMPAQ/HP, AND FREESCALE)

DESIGNATED UNDER THE PROTECTIVE ORDER ENTERED IN THIS CASE

TABLE OF CONTENTS

PAGE

TABLE OF CONTENTS
(CONTINUED)

PAGE

TABLE OF CONTENTS
(CONTINUED)

PAGE

TABLE OF CONTENTS
(CONTINUED)

PAGE

TABLE OF CONTENTS
(CONTINUED)

PAGE

I. INTRODUCTION

1. My name is Robert P. Colwell My address is 3594 N.W. Bronson Crest Loop, Portland, Oregon, 97229. I am over eighteen years of age and would otherwise be competent to testify to the matters in this report, if I am called upon to do so at trial.

2. I have been retained by Cooley Godward LLP on behalf of Defendant NVIDIA Corporation ("NVIDIA") to serve as an expert in the above-referenced matter.

3. For purposes of this report, I have been asked to provide an expert technical analysis and opinion as to whether the asserted claims of U.S. Patent No. 5,710,906 ("the '906 patent") are invalid. My opinions are based upon the information I have considered, received, and reviewed to date. My opinions may be supplemented or modified if I receive additional information, and may also be supplemented to reply to additional information that arises at trial or that is provided by OPTi, Inc. ("OPTi") or witnesses retained by OPTi.

II. QUALIFICATIONS

4. I received a Ph.D. in Computer Engineering in 1985, and an M.S.E.E. in Computer Engineering in 1978, from Carnegie-Mellon University. I received a B.S. in Electrical Engineering from the University of Pittsburgh in 1977.

5. From 1977-1980, I worked as a hardware engineer at Bell Telephone Labs in Holmdel, New Jersey working on 8-bit and 32-bit microprocessors. From 1981-1984, I worked as a hardware engineer for Perq Systems in Pittsburgh, Pennsylvania designing graphics-display hardware for first-generation engineering workstations. From 1985-1990, I worked as a hardware architect for Multiflow Computer in New Haven, Connecticut, where I was one of the seven hardware designers who created the world's first VLIW (very long instruction word) scientific supercomputer.

6. In 1990, Intel Corporation hired me as a senior CPU architect charged with designing Intel microprocessors. From 1990-1992, I was one of the three senior architects responsible for designing Intel's P6 microprocessor architecture, which is the processing core of

the well-known Pentium II, Pentium III, Celeron, Xeon, and Centrino families of Intel microprocessors. These microprocessors are commonly found in laptop, desktop, and server computer systems.

7. From 1992-2001, I was the lead Intel IA32 architect responsible for all of Intel's CPU architecture development efforts. My responsibilities included initiating and leading the development of Intel's Pentium 4 CPU core development, during which time I managed up to 100 other Intel engineers and employees. I was named an Intel Fellow in 1997. At that time, I was one of only 27 Intel Fellows out of approximately 80,000 Intel employees.

8. Since leaving Intel in 2001, I have been a consultant to industry and academia on various computer hardware and software issues. I have also written numerous publications, magazine articles, and authoritative texts in the field of CPU design and computer architecture. These publications include the P6 chapter of "Superscalar Processor Design" by Shen & Lipasti, published by McGraw-Hill in 2003, and numerous articles in IEEE Computer Magazine and other technical publications. I also frequently present talks at various symposia regarding CPU architecture and design, as well as computer systems in general.

9. I am a named inventor on numerous patents, including 35 patents that I worked on while at Intel. My work experience, patents, and awards are covered in more detail in my curriculum vitae, which is attached as Attachment A.

III. PRIOR TESTIMONY AND COMPENSATION

10. A list of my prior deposition and trial testimony over the last four years is attached as Attachment A. I am being compensated for my time on this case at a rate of $250 per hour.

IV. MATERIALS REVIEWED

11. The materials that I have considered in connection with this report include, in addition to all material cited and referenced in the body of my report and in my charts and timelines:

- The OPTi presnoop patents '906, '036, and '291 patents. (EX. 17, 18, 19).
- The Prosecution Histories of the OPTi presnoop patents. (EX. 20, 21, 22).
- PCI Local Bus Specification Revision 2.0 (April 30, 1993) (EX. 64).
- Intel, CDC Cache & DRAM Controller Target Specification, Rev. 1.0 (EX. 122).
- Intel CDC Cache & DRAM Controller Target Specification, Rev. 0.2 (EX. 123).
- The Intel '094 Patent (EX. 96).
- Triton Host Memory/Cache Controller/PCI Bridge, Preliminary Component Specification Draft (EX. 98).
- The Triton System Controller (TSC) and Triton Data Path (TDP) Component Specification (EX. 99).
- Intel Invention Disclosure of Hayek, Oztaskin, Young, and Langendorf (EX. 100).
- Report of Barry L. Kramer re: Intel Triton analysis (Attachment B) (I also conferred with Mr. Kramer about his analysis).
- Report of Barry L. Kramer re: Compaq Prosignia 500 analysis (Attachment C) (I also conferred with Mr. Kramer about his analysis).
- Report of Gerald Mossinghoff.
- Deposition Transcript of Nadir Amini (plus exhibits).
- Deposition Transcript of Raj Jaswa (plus exhibits).
- Deposition Transcript of Ali Oztaskin (plus exhibits).
- Deposition Transcript of Bruce Young (plus exhibits).
- Deposition Transcript of Brian Langendorf (plus exhibits).
- Deposition Transcript of George Hayek (plus exhibits).

- Deposition Transcript of Michael Garcia (plus exhibits).
- Deposition Transcript of Christopher Bryant (plus exhibits).
- Deposition Transcript of Brian Reynolds (plus exhibits).
- Deposition Transcript of Michael Collins (plus exhibits).
- Deposition Transcript of Subir Ghosh (plus exhibits).
- Deposition Transcript of Robert Harris (plus exhibits).
- Deposition Transcript of Thomas Shanley (plus exhibits).
- Deposition Transcript of Gary Thome (plus exhibits).
- Deposition Transcript of Ron Epstein (plus exhibits).
- Deposition Transcript of Bernard Marren (plus exhibits).
- Yong Yao, Vendors fight for Pentium core-logic market; Intel is king of the land, most others are losing chip-set share (Cover story), Microprocessor Report (August 21, 1995) (NV 021342-021348).
- Hot Chips VI Symposium Record (SMITH000047-269).
- Viper-M Data Book (OPTINVIDIA 007132-007389).
- Plaintiff Opti Inc.'s Objections and Responses to Defendant NVIDIA's First Set of Interrogatories (EX. 162).
- Plaintiff Opti Inc.'s Objections and Responses to Defendant NVIDIA's Second Set of Interrogatories.
- *Electronic News: Intel readies 'Triton' PCI chipset - Intel's 82430FX - Product Announcement* (Jan. 30, 1995) (NV040393-40394).

12. Throughout this report, I have cited excerpts from depositions and exhibits that have been identified in this case. The citations I provide here are meant to be illustrative, and I may refer to additional portions of the testimony and/or exhibits as necessary and appropriate at trial. In addition, the materials that I plan to use as exhibits in this case will primarily be selected from these materials, although I may later reference exhibits identified by other experts. Additionally, I expect that certain demonstrative evidence, including, but not

limited to, charts, graphs, and drawings, which have not yet been prepared, will be used during my testimony.

13. The charts and timelines that accompany this report are intended to be read together with the body of this report. The inclusion of certain material in the claim charts and/or timelines is not mean to exclude in any way the analysis and opinions set forth in this report. The claim charts and timelines are meant to summarize and augment the opinions that are set forth herein.

V. ANALYSIS OF COURT'S CLAIM CONSTRUCTION RULING

14. I have been given the following instruction regarding means plus function limitations:

> **7.4 MEANS-PLUS-FUNCTION CLAIM LIMITATIONS**
>
> Some patent claim limitations may describe a "means" for performing a function, rather than describing the structure that performs the function. For example, let's say that a patent describes a table in which the legs are glued to the tabletop. One way to claim the table is to recite the tabletop, four legs and glue between the legs and the tabletop. Another way to claim the table is to recite the tabletop and the legs, but, rather than recite the glue, recite a "means for securing the legs to the tabletop". This second type of claim limitation is called a "means-plus-function" limitation. It describes a means for performing the function of securing the legs to the tabletop, rather than expressly reciting the glue.
>
> When a claim limitation is in means-plus-function form, it covers the structures described in the patent specification for performing the function stated in the claim, and also any structure that is equivalent to the described structures. In our example, the claim covers a table using glue to secure the legs to the tabletop, as described in the patent, and any equivalent structure that performs the function of securing the legs to the tabletop.
>
> Claims ___ of the ___ patent include means-plus-function limitations. In instructing you about the meaning of a means-plus-function claim limitation, I will tell you, first, the function that each of the means-plus-function claim limitations performs; and second, the structure disclosed in the patent specification that corresponds to each means-plus-function limitation.
>
> [Fed. Cir. Bar Assn. Model J. Instr. (2002)].

15. I have also reviewed the Court's claim construction for the presnoop patents. Most of the Court's rulings do not require interpretation for purposes of this invalidity

report. However, the Court's ruling on the "means for initiating a next-line inquiry" requires interpretation. The Court held that the means for initiating a next-line inquiry is "the logic circuitry of the SYSC/IPC schematically illustrated in Figure 9 of the patent."

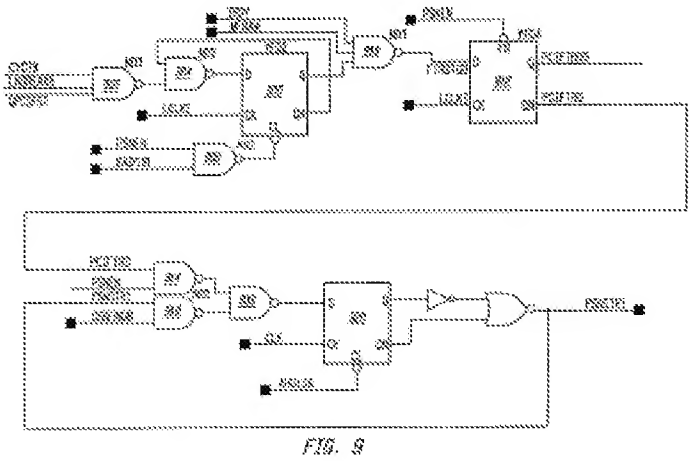


FIG. 9

[EX. 17, '906 Patent].

16. The patent describes Fig. 9 as follows:

> FIG. 9 is a schematic diagram of circuitry in the system controller 116 which produces the PSNSTR1 signal used in FIG. 8. As previously mentioned, PSNSTR1 carries a high-going pulse when it is desired to initiate a predictive snoop cycle during a PCI master burst transfer.
>
> Referring to FIG. 9, a three-input NAND gate 902 receives a QPCIFST signal, which is high during the first transfer of a PCI burst or the beginning of a new cache line transfer.
>
> Another input of NAND gate 902 receives a CYCTX signal, which is asserted when both IRDY# and TRDY# are sampled active (a transfer is occurring). NAND gate 902 also receives an LNBREAKB signal, which is low only if the data unit then being transferred is the highest data unit in a cache line. Accordingly, the output of NAND gate 902 will go low during the transfer of the first data unit to be transferred from a line of secondary cache, but not if the transfer is beginning with the highest data unit in the line of secondary memory. This is consistent with the discussion above with respect to FIG. 6 in which predictive snoop is omitted in this situation.
>
> The output of NAND gate 902 is connected to one input of a two-input NAND gate 904, the output of which is connected to the D input of a flip-flop 906. The QN output of flip-flop 906 is connected back to the second input of NAND gate 904. The flip-flop 906 has an inverting clear input which is connected to the output of an AND gate 908, one input of which receives PSNEN, which remains high at all times pertinent herein, and the other input of which receives an EADS1B signal. EADS1B goes low

after EADS#, thereby clearing flip-flop 906. Accordingly, flip-flop 906 latches the output of NAND gate 902 until after EADS# has been asserted.

The Q output of flip-flop 906 is inverted and qualified, in three-input NAND gate 910, by IRDY and MFRAME. IRDY is the inverse of the PCI-bus 118 IRDY#signal, and as previously explained, MFRAME essentially follows the inverse of the PCI-bus FRAME# signal. Thus, NAND gate 910 blocks the output of flip-flop 906 if the PCI device 138 has already indicated that the present transfer is to be the last transfer of the burst. Otherwise, the output of NAND gate 910 (called FTRDTGB ("first TRDY# trigger")) carries a one PCICLK-wide low-going pulse, beginning with the PCICLK rising edge that ends the first PCI transfer of the current line of secondary memory.

The output of NAND gate 910, FTRDTGB, is connected to the D input of a flip-flop 912, which is clocked on LCLKI. Flip-flop 912 thus delays FTRDTGB by one PCICLK to enable other circuitry (not shown) in the system controller 116 to increment the secondary memory line address on HA(31:5) (FIG. 1).

The QN output of flip-flop 912, designated PCIFTRD, is connected to one input of a two-input NAND gate 914, the other input of which receives PSNEN. The output of NAND gate 914 is connected to one input of a two-input NOR gate 916, the other input of which receives the output of another NAND gate 918. One input of NAND gate 918 receives a CSNENDB signal, which is high until EADS# is asserted, and the other input of NAND gate 918 receives the PSNSTR1 signal. The output of NAND gate 916 is connected to the D input of a flip-flop 920 which is clocked by CLK (equivalent to the host bus clock signal HCLK). The QN output is NORed with an inverted version of the Q output of flip-flop 920 to produce the PSNSTR1 signal, which is fed back to NAND gate 918. PSNSTR1 therefor carries a high-going pulse which is synchronized with the host bus clock signal HCLK#, and which remains high until EADS# is asserted.

As previously described, PSNSTR1 is provided to an input of NAND gate 822 in FIG. 8 and, like LT2, initiates an L1 cache inquiry cycle.

[EX. 17, '906 Patent Col. 23:32-24:35].

17. Figure 9 of the '906 patent shows circuitry for asserting the PSNSTR1 signal. The pre-snoop patents explain that "PSNSTR1 will carry a high-going pulse when it is desired to assert EADS# for predictive snoop cycle later in the burst." Thus, the PSNSTR1 signal is initiated in order to initiate the EADS# signal and initiate a next-line inquiry.

18. In order to initiate a next-line inquiry, the secondary memory line address on HA(31:5) must be incremented to point to the next-line. The pre-snoop patents note that the address HA(31:5) must be incremented, but state that the structure to do so is "not shown":

> The output of NAND gate 910, FTRDTGB, is connected to the D input of a flip-flop 912, which is clocked on LCLKI. Flip-flop 912 thus delays FTRDTGB by one PCICLK to enable other circuitry (*not shown*) in the system controller 116 to increment the secondary memory line address on HA(31:5) (FIG. 1).

[EX. 17, '906 Patent Col. 24:10-15 (emphasis added)].

19. The PSNSTR1 signal is asserted when IRDY, FRAME, and TRDY are asserted for the first data unit of the current line, the PSNEN signal (presnoop enabled) is asserted as well, and the LNBREAKB signal does not indicate that the transfer is starting on the last data unit of the current line. Thus, the structure in FIG. 9 shows that the PSNSTR1 signal is asserted when the first data unit of the current line is being transferred on the PCI bus (the assertion of TRDY). In particular, the patent indicates that the PSNSTR1 signal depends on the FTRDTGB signal, which is the "first TRDY# trigger."

> Otherwise, the output of NAND gate 910 (called FTRDTGB ("first TRDY# trigger")) carries a one PCICLK-wide low-going pulse, beginning with the PCICLK rising edge that ends the first PCI transfer of the current line of secondary memory.

[EX. 17, '906 Patent Col. 24:5-9].

20. My analysis of the means-plus-function claims for the OPTi presnoop patents, set forth below and in the accompanying charts, will be based on this analysis of FIG. 9 above.

VI. BACKGROUND INFORMATION.

21. I understand that OPTi will present its evidence and witnesses at trial first, before NVIDIA presents its witnesses, and that by the time I testify, the jury should already be familiar with the basic technical terms and concepts in the case. Nevertheless, to the extent necessary, I expect to present summary basic background information regarding the technology,

including how microprocessors work, what a peripheral bus is, how DRAM works, how write-back caches and write-through caches work, what "snooping" is, how cache coherency is maintained, how the PCI Local Bus specification commands work (particularly focusing on MRM), how PCI signals operate (e.g., FRAME, IRDY, TRDY, EADS), how PCI data transfers work (e.g., address phase, data phase), how the PCI Local Bus specification includes limitations on wait states, as well as what prefetching is. Such information will be similar to the information already presented in the tutorials to the Court by both parties, and would be similar to the description of background technology presented in the prior art cited herein. I incorporate NVIDIA's technical tutorial by reference as a basic summary of the background technology that I may provide.

VII. PERSON OF ORDINARY SKILL IN THE ART.

22. I have been asked to give an opinion regarding the level of skill of a person of ordinary skill in the art. I have been given the following instruction regarding the level of skill in the art:

> 10.9.3 LEVEL OF ORDINARY SKILL
> Obviousness is determined from the perspective of a person of ordinary skill in the art. This person is presumed to know all of the prior art, not just what the inventor may have known. When faced with a problem, this ordinary skilled person is able to apply his or her experience and ability to the problem and also to look to any available prior art to help solve the problem.
> Factors to consider in determining the level of ordinary skill in the art include the educational level and experience of people working in the field, the types of problems faced by workers in the art and the solutions found to those problems, and the sophistication of the technology in the field.

[Fed. Cir. Bar Ass'n Model Jury Inst. (2002)].

23. The field of the presnoop patents was relatively narrow and specialized. Relatively few persons were working on chipsets for PCs at the time. Chipset makers included Intel, OPTi, VLSI, UMC, and VIA, and SiS, as well as some PC OEMs, including Compaq and IBM. I understand, from OPTi's interrogatory response, that OPTi contends that its date of invention for the presnoop patent claims is May 1994. At that time, a person of ordinary skill in

the art in the field of the presnoop patents (PC chipset design) would have had practical experience in chipset design, and in particular, knowledge and understanding of the PCI local bus specification, Rev. 2.0. A person of ordinary skill in the art would have had at least an undergraduate EE degree and some practical experience in chipset design. Due to the fast-moving nature of the chipset design field at the time, and due to the fact that virtually all chipset work was done in industry and in a private setting, academic experience alone would only qualify a person for an entry-level position in the field. Those of ordinary skill in the chipset design art would have had practical experience. The witnesses working in the field who have testified in this case generally had practical working experience.

VIII. INSTRUCTIONS RE: LEGAL STANDARDS

A. Burden of proof instruction.

24. For background on the law, I have been given the following instruction regarding the burden of proof:

> **1.1 BURDENS OF PROOF**
>
> In any legal action, facts must be proved by a required standard of evidence, known as the "burden of proof". In a patent case such as this, there are two different burdens of proof that are used.
>
> The first burden of proof standard requires that, in order for a party to prevail, you must be persuaded that what the party seeks to prove is more probably true than not true.
>
> The second burden of proof standard is a higher one. It requires that you must be persuaded that it is highly probable that what the party seeks to prove is true.
>
> You may have heard of a burden of proof that is used in criminal cases called "beyond a reasonable doubt". That requirement is the highest burden of proof. It does not apply to a patent case such as this one, and you should, therefore, put it out of your mind.
>
> I will now give you some background about the nature of this case and the issues you will be deciding. For each issue, I will instruct you as to the burden of proof that will apply. At the end of the trial I will review for you the burden of proof to apply, either the more probable than not standard or the highly probably standard, to each issue in this case.
>
> [Fed. Cir. Bar Ass'n Model Jury Instructions (2002)].

B. Anticipation instructions.

25. I have been asked to give an opinion regarding anticipation by the prior art. I have been given the following instruction regarding the law of anticipation:

> 10.8 ANTICIPATION/LACK OF NOVELTY
>
> A person cannot obtain a patent on an invention if someone else has already made the same invention. In other words, the invention must be new. If an invention is not new, we say that it was "anticipated" by the prior art. An invention that is "anticipated" by the prior art is not entitled to patent protection. A party challenging the validity of a patent must prove anticipation by the highly probable standard.
>
> In order for a patent claim to be anticipated by the prior art, each and every limitation of the claim must be present within a single item of prior art, whether that prior art is a publication, a prior patent, a prior invention, a prior public use or sale, or some other item of prior art. You may not find that the prior art anticipates a patent claim by combining two or more items of prior art.
>
> A printed publication or patent will not be an anticipation unless it contains a description of the invention covered by the patent claims that is sufficiently detailed to teach a skilled person how to make and use the invention without undue experimentation. That means that a person skilled in the field of the invention reading the printed publication or patent would be able to make and use the invention using only an amount of experimentation that is appropriate for the complexity of the field of the invention and for the level of expertise and knowledge of persons skilled in that field.
>
> In deciding whether or not a single item of prior art anticipates a patent claim, you should consider that which is expressly stated or present in the item of prior art, and also that which is inherently present. Something is inherent in an item of prior art if it is always present in the prior art or always results from the practice of the prior art, and if a skilled person would understand that to be the case.
>
> A prior public use by another may anticipate a patent claim, even if the use was accidental or was not appreciated by the other person. Thus, a prior public use may anticipate an invention even if the user did not intend to use the invention, or even realize he or she had done so.
>
> In this case, defendant contends that claims ___ of the ___ patent are invalid because they are anticipated by _____. If you find that [defendant] has proved that it is highly probable that claims ___ are anticipated, then you must find that the claims are invalid.

C. Prior invention instructions.

26. I have been provided the following summaries of the law of invalidity, which I understand have been taken directly from Model Jury Instructions. The following

description of prior invention was provided to me from the Federal Circuit Bar Association Model Jury Instructions:

> PRIOR ART -- PRIOR INVENTION
> An invention made by another person before the inventor on the patent made the invention is prior art to the patent claim, unless that other person abandoned, suppressed or concealed his or her invention.
> As a general rule, the first person to reduce an invention to practice is said to be the first inventor. An invention is reduced to practice either when a patent application is filed or when the invention is made and shown to work for its intended purpose. Thus, if another person reduces to practice an invention before the inventor on the patent, then the reduction to practice by the other person will be prior art to the patent claims.
> Let's consider an example. Mr. Smith has a patent on a table. He reduced his table to practice on April 1. Ms. Jones invents the same table. She built her table on March 1, one month before Mr. Smith reduced his table to practice. Ms. Jones' invention of the table is prior art to Mr. Smith's patent
> claims because Ms. Jones reduced her table to practice one month before Mr. Smith's reduction to practice.
>
> There is, however, an important exception to this general rule. Someone who was first to conceive of an invention but reduced it to practice after someone else will be the first inventor if he or she was the first to conceive of the invention and he or she exercised "reasonable diligence" in reducing the invention to practice from a time just before the other person's conception. Conception of an invention occurs when the inventor has formed the idea of how to make and use every aspect of the patented invention, and all that is required is that it be made, without the need for any further inventive effort. Reasonable diligence means that the inventor worked continuously on reducing the invention to practice. Interruptions necessitated by the everyday problems and obligations of the inventor or those working with him or her do not prevent a finding of diligence.
>
> Let's change our example slightly. Mr. Smith conceived of his table on February 1 and reduced it to practice on April 1. Ms. Jones conceived of the table on January 1, one month before Mr. Smith's conception, and built it on May 1, one month after Mr. Smith's reduction to practice. If Ms. Jones was reasonably diligent in building the table from the time just before Mr. Smith's February 1 conception up to the time that she built the table on May 1, she is the first inventor of the table and her invention is prior art to Mr. Smith's patent claims. The final requirement for a prior invention to be prior art is that the prior inventor did not abandon, suppress or conceal his or her invention. Generally, an invention was not abandoned, suppressed or concealed if the invention was made public, sold or offered for sale, or otherwise used.

27. The following was provided to me from the AIPLA model jury instructions:

> Prior Invention
> [The defendant] contends that claim[s] _____ of the _____ patent was anticipated because the invention defined in [that claim] [those claims] was invented by another person [X], before [the patentee] invented his invention.
> A patent claim is invalid if the invention defined by that claim was invented by another person in the United States, before it was invented by the patentee, and that other person did not abandon, suppress or conceal the invention.
>
> To show anticipation of the [patented] invention, [the defendant] must show by clear and convincing evidence that before [the patentee] invented his invention, [X] actually built a device [practiced a process] [made a composition] that included all of the elements of claim ___ of the ___ patent. In addition, [the defendant] must show that [X]'s device was sufficiently developed that one skilled in the art would have recognized that it would work for its intended purpose. To be an anticipation, all of those acts must have been completed before _____ [the inventor of the patent in suit] invented the invention defined by claim ___ of the ___ patent. [If the prior invention was abandoned, suppressed or concealed, it does not anticipate the ___ patent] However, it is not necessary that [the patentee] had knowledge of that prior invention.
>
> In cases where priority of invention is an issue to be submitted to the jury, further instructions will be required. For example, the jury will need to consider not only the dates when the respective inventions were conceived, but also when the inventions were reduced to practice. An inventor who claims to be the first to conceive of a prior invention but was the last to reduce to practice must also show reasonable diligence from a time just before the other party entered the field until his own reduction to practice in order for the "prior invention" to anticipate the claimed invention in suit.

D. Obviousness instructions.

28. I have been asked to give an opinion regarding obviousness of the presnoop claims in suit. I have been given the following instruction regarding the law of obviousness.

> 10.9 OBVIOUSNESS
> As I mentioned earlier, an inventor is not entitled to a patent if his or her invention would have been obvious to a person of ordinary skill in the field of the invention at the time the invention was made.

Unlike anticipation, obviousness may be shown by considering more than one item of prior art. The question is, would it have been obvious for a skilled person who knew of the prior art to make the claimed invention? If the answer to that question is yes, then the patent claims are invalid. [Defendant] has the burden of proving by the highly probable standard that claims ___ of the ___ patent are invalid for obviousness.

Obviousness is determined from the perspective of a person of ordinary skill in the field of the invention. The issue is not whether the claimed invention would have been obvious to you, to me as a judge, or to a genius in the field of the invention. Rather, the question is whether or not the invention would have been obvious to a person of ordinary skill in the field of the invention.

In deciding obviousness, you must avoid using hindsight; that is, you should not consider what is known today or what was learned from the teachings of the patent. You should not use the patent as a road map for selecting and combining items of prior art. You must put yourself in the place of a person of ordinary skill at the time the invention was made.

You must also keep in mind that the test for obviousness is not whether or not it would have been obvious to try to make the invention, but rather whether or not the invention would have been obvious to a person of ordinary skill in the inventor's field at the time the invention was made.

In determining whether or not these claims would have been obvious, you should make the following determinations:

First, what is the scope and content of the prior art?

Second, what differences, if any, are there between the invention of the claims of the patent and the prior art?

Third, what was the level of ordinary skill in the art at the time the invention was made?

Fourth, are there any objective indications of non-obviousness?

[Fed. Cir. Model Jury Inst. (2002).]

10.9.1 THE SCOPE AND CONTENT OF THE PRIOR ART
Determining the scope and content of the prior art means that you should determine what is disclosed in the prior art relied on by [defendant]. You must decide whether this prior art was reasonably relevant to the particular problem the inventor faced in making the invention covered by the patent claims. Such relevant prior art includes prior art in the field of the invention, and also prior art from other fields that a person of ordinary skill would look to when attempting to solve the problem.

10.9.2 DIFFERENCES BETWEEN THE INVENTION OF
THE CLAIMS AND THE PRIOR ART

In determining the differences between the invention covered by the patent claims and the prior art, you should not look at the individual differences in isolation. You must consider the claimed invention as a whole and determine whether or not it would have been obvious in light of all of the prior art.
In deciding whether to combine what is described in various items of prior art, you should keep in mind that there must be some motivation or suggestion for a skilled person to make the combination covered by the patent claims. You should also consider whether or not the prior art "teaches away" from the invention covered by the patent claims. The question to be answered is: Would someone reading the prior art be discouraged from following the path taken by the inventor?

10.9.3 LEVEL OF ORDINARY SKILL
Obviousness is determined from the perspective of a person of ordinary skill in the art. This person is presumed to know all of the prior art, not just what the inventor may have known. When faced with a problem, this ordinary skilled person is able to apply his or her experience and ability to the problem and also to look to any available prior art to help solve the problem.
Factors to consider in determining the level of ordinary skill in the art include the educational level and experience of people working in the field, the types of problems faced by workers in the art and the solutions found to those problems, and the sophistication of the technology in the field.

10.9.4 OBJECTIVE INDICATIONS CONCERNING OBVIOUSNESS
You also must consider what are referred to as objective indications of non-obviousness. Some of these indications of non-obviousness are:
1. Commercial success of products covered by the patent claims or made by a process covered by the patent claims.
2. A long-felt need for the invention.
3. Failed attempts by others to make the invention.
4. Copying of the invention by others in the field.
5. Unexpected results achieved by the invention.
6. Praise of the invention by the infringer or others in the field.
7. The taking of licenses under the patent by others.
8. Expressions of surprise by experts and those skilled in the art at the making of the invention.
9. The patentee proceeded contrary to the accepted wisdom of the prior art.
The presence of any of these objective indications may suggest that the invention was not obvious. These objective indications are only relevant to obviousness if there is a connection, or nexus, between them and the invention covered by the patent claims. For example, commercial success is relevant to obviousness only if the success of the product is related to a feature of the patent claims. If the commercial success is the result of something else, such as innovative marketing, and not to a patented feature, then you should not consider it to be an indication of non-obviousness.

10.9.5 DETERMINATION OF OBVIOUSNESS
[Defendant] contends that the invention claimed in claims ___ of the ___ patent would have been obvious to a person of ordinary skill in the field of the invention at the time the invention was made in light of the _____ item of prior art combined with _____. If you find that [defendant] has proved obviousness by the highly probable standard, then you must find that the claims are invalid for obviousness.

IX. SUMMARY OF OPINIONS.

A. The asserted claims of the OPTi '906 presnoop patent are invalid in view of the prior invention of Intel employees, including Bruce Young, Ali Oztaskin, and Brian Langendorf (and also published in Shanley).

B. The asserted claims of the OPTi '906 patent are invalid in view of the prior invention of Compaq employees, including Michael Collins and Gary Thome.

35. Claims 1, 7, 8, 9, 21, and 26 of the '906 Patent are anticipated under § 102 and/or rendered obvious under § 103 by the prior invention of the snoop ahead method by Compaq employees, including Gary Thome and Michael Collins.

36. The Compaq employees conceived of their invention in late 1992 / early 1993 and reduced it to practice in silicon by February 1994. (OPTi's first conception document is dated May 30, 1994.) The Compaq employees did not abandon, suppress, or conceal their invention, as Compaq filed a patent application in October 1994 (OPTi's filing is July 1995) and released a commercial chipset in November 1994.

C. The asserted claims of the OPTi '906 patent are invalid in view of the prior invention of Motorola employees, including Michael Garcia, Christopher Bryant, and Brian Reynolds.

37. Claims 1, 7, 8, 9, 21, and 26 of the '906 Patent are anticipated under § 102 and/or rendered obvious under § 103 by the prior invention of the snoop ahead method by Motorola employees, including Michael Garcia, Christopher Bryant, and Brian Reynolds.

38. The Motorola employees conceived of their invention in late 1992 / early 1993 and reduced it to practice in silicon by May 22, 1994. (OPTi's first conception document is dated May 30, 1994.) The Motorola employees did not abandon, suppress, or conceal their invention, as Motorola published several papers regarding the invention and released a commercial chipset in the fall of 1994.

D. The asserted claims of the OPTi presnoop patents are invalid in view of the prior invention of IBM (Microchannel and Amini).

39. Claims 1, 7, 8, 9, 21, and 26 of the '906 Patent are anticipated under § 102 and/or rendered obvious under § 103 by the prior invention of the snoop ahead method by IBM employees, including Nadir Amini, as set forth in the Amini patent, U.S. Patent No. 5,581,714. [EX. 215].

40. Claims 1, 7, 8, 9, 21, and 26 of the '906 Patent are rendered obvious under § 103 by IBM's Micro Channel architecture described in the IBM TDB.

X. DETAILED DESCRIPTION OF AND BASIS FOR OPINIONS.

A. The asserted claims of the OPTi patents are invalid in view of the prior invention by Mr. Young, Mr. Oztaskin, Mr. Hayek, and Mr. Langendorf (including the Shanley publication).

192. Incidentally, snoop ahead isn't intended to speed up the uncommon case that the cache contains dirty data. Snoop ahead is supposed to speed up the more common case, where the snoops don't actually hit anything in a cache– when this happens, PCI gets to run at full speed for extended periods, and that's where the performance gain exists.

193. Ghosh "didn't pay much attention" to whether anyone else had addressed the same problem. He testified:

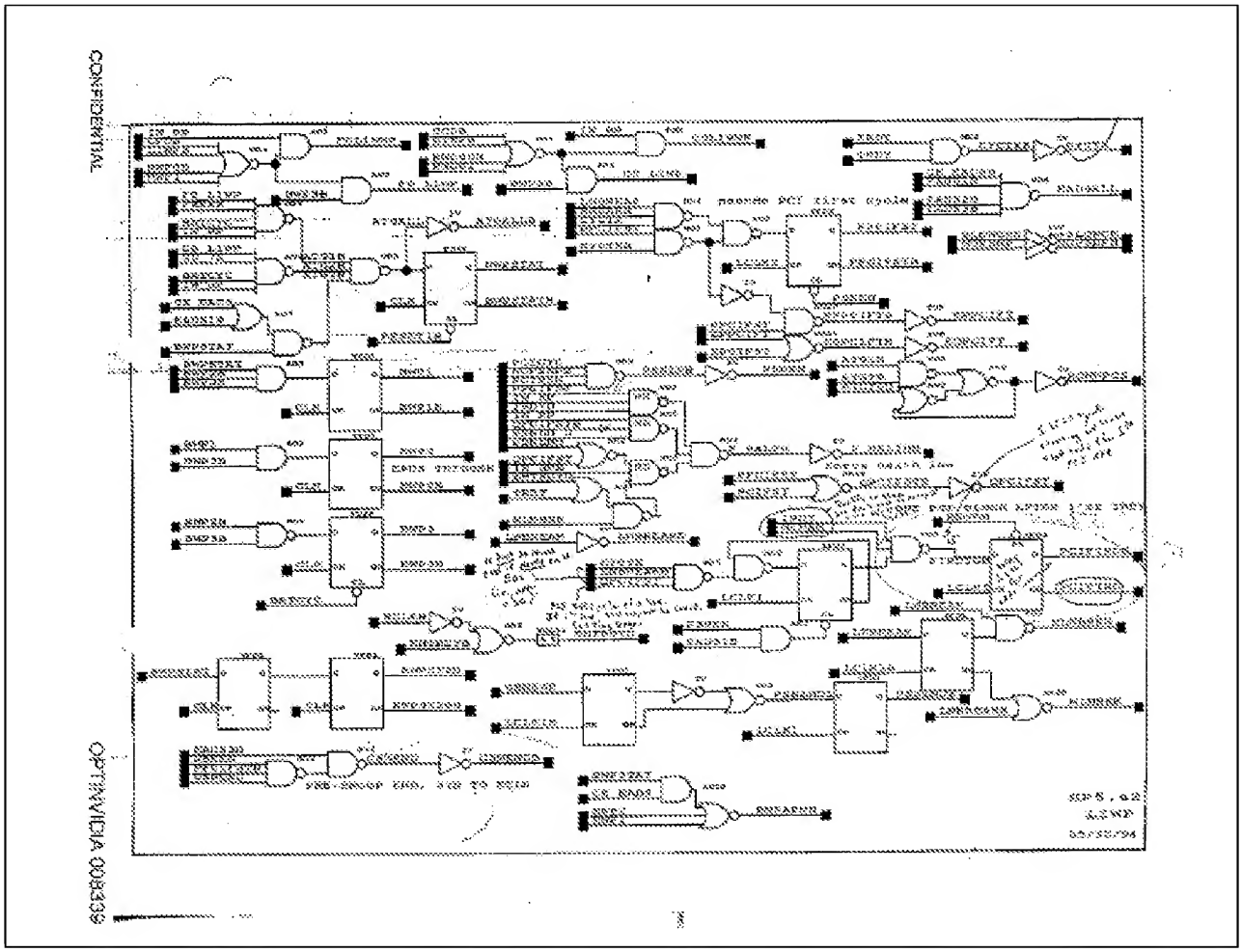
37

Q. Is it accurate that you identified the performance bottleneck problem you just mentioned when you were reading the PCI spec?
A. There were many specs that I had to go through, including PCI spec.
Q. But you identified this problem on your own, no one told you about it?
A. That is correct.

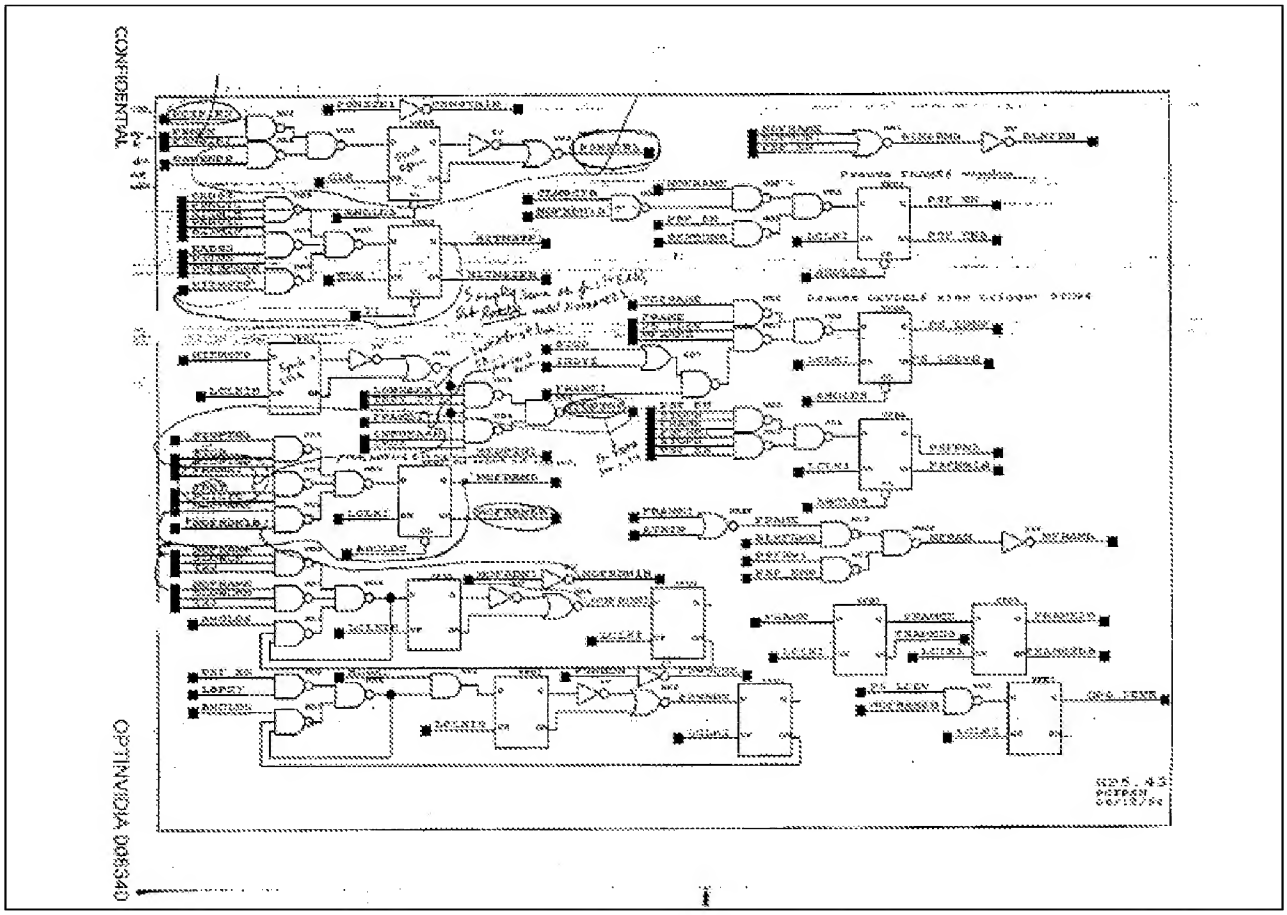
38

Q. At the time, when you identified this performance bottleneck problem in a burst access, did you wonder whether anyone else had identified this same problem?
MR. GRACEY: Object to form.
You can answer.
THE WITNESS: I didn't pay much attention to it.

195. OPTi decided to do the snoop ahead protocol, which Shanley had published earlier. The first dated document showing that OPTi was thinking about presnoop is dated May 30, 1994. [EX. 41, below]. The page below, dated May 30 shows half of FIG. 9 of the '906 patent.



196. The other half of '906 FIG. 9 is dated June 15, 1994. [EX. 41].


CONFIDENTIAL
OPTIMOSA 003340

228. If OPTi had invented a chipset that could transfer data units without intervening wait states, it would have disclosed it in the OPTi patent application filed in July 1995, as the "best mode" of practicing presnoop.

B. The asserted claims of the OPTi presnoop patents are invalid in view of the prior invention of Compaq employees, including Michael Collins and Gary Thome.

244. The asserted claims of the OPTi presnoop patent are invalid in view of the prior invention of certain Compaq employees, including Michael Collins and Gary Thome, and described in U.S. Patent No. 5,634,073, filed October 14, 1994 and commercialized in the Triflex/PCI product sold in November 1994.

245. The chipset that Mr.Collins and Mr. Thome invented was called the CMC/DMC chipset. Mr. Collins and Mr. Thome worked at Compaq in Houston, Texas, where they invented the CMC/DMC chipset. Mr. Thome was in charge of the entire project. Mr. Collins was project manager of the North Bridge. Mr. Collins testified:

14

Q If you see on the front of Exhibit 182, which is the '073 patent, there are listed five different inventors?
A Yes.

Q You were the first listed inventor, correct? 1:55:58PM
A Yes.
Q What was your role in creating the triflex architecture?
A I was the project manager of this particular north bridge, so of the CMC project I was the project manager. 1:56:09PM

15

Q What was Gary Thome's role? 1:56:11PM
A Gary Thome did, he was in the product development group so he would be functioning as the total project. So being that this chipset would go into a product, he would be responsible for the total project and I was responsible for the particular north bridge. 1:56:26PM

246. As cited below and described in detail, Mr. Thome and Mr. Collins both testified that the CMC/DMC chipset both prefetched and presnooped. Mr. Collins noted that in the Compaq system, prefetching and presnooping went together:

55

Q Let me ask this question: Do you understand what is referred to by the term "presnoop"?
A Yes.
Q What is your understanding of that term? 3:06:16PM
A Presnooping was, again, it was speculative. And I'm not sure if it was coming from the PCI spec or it was in one of the patents, to be honest with you. Presnooping is when you are reading ahead, you're speculatively asking for the next cache line. That next cache line has to always be snooped because that could be a modified hit in the caching. 3:06:37PM

56

So prefetching and presnooping really go together, or read ahead. So read multiple, read ahead, prefetching, presnooping, they are all very similar to the same concept. 3:06:44PM 3:07:01PM
Q So did the triflex PCI architecture as described in the '073 patent implement presnooping?
A I believe so. 3:07:20PM
Q Did it implement it along with prefetching?
A Yes.

247. OPTi contends that it conceived and reduced the presnoop patents to practice as follows:

> Pre-Snoop Patents
>
> (a) U.S. Patent No. 5,710,906 ("the '906 Patent")
>
> The inventions set forth in the asserted claims were conceived sometime prior to May 30, 1994. A working prototype was produced in October of that year and was successfully demonstrated sometime prior to November 12, 1994. Between May 30, 1994 and November 12, 1994 the inventors worked with due diligence to reduce their conceived invention to practice. The first logic diagrams showing the Pre-Snoop circuitry were completed on May 30, 1994. Logic diagrams showing the circuitry disclosed in the patent were completed July 30, 1994. A prototype chip embodying the invention was ordered on August 23, 1994. In the normal course, the prototype would have been received in late October or early November of 1994. The prototype was demonstrated to be operational no later than November 12, 1994. The initial chipsets shipped shortly thereafter. Documents relating to the conception and reduction to practice of the inventions set forth in the asserted claims have been produced with OPTi's P.R. 3-1 Disclosures and Preliminary Infringement Contentions.

[EX. 162, OPTi's Response to NVIDIA's Interrogatory No. 2].

248. Mr. Jaswa, a former OPTi founder who was a VP at the time, testified about the development process at OPTi as follows:

> 63
>
> Do you recall OPTi releasing a chipset in early '95?
>
> A. Yes, yes.
>
> Q. And do you recall that that chipset had this presnoop feature included in it?
>
> A. Kind of remember, yes.
>
> Q. Nine months. So if we wanted to know when development of that chipset started, would we go nine months prior to the date that it was released?
>
> 64

A. That's correct, yes.
Q. Was that typical for all of OPTi's chipsets?
MR. GRACEY: Object to form.
THE WITNESS: Yes. As -- we used to have something called ASICS and we used to have something called, you know, what we call -- okay. We used to have gate array-based technology and then we used to have fully kind of custom, laid-out chips. Custom, laid-out chips, you've got to add another six months to them. So if it is a gate array implementation, then six to nine months is -- six months for development and three months for prototype is the normal development cycle.
Q. How would you figure out whether a chipset was gate array or the other type?
A. It depended on kind of gig [sic, "gate"]count. And in those days if the gate count was under -- in the mid '90s if the gate count was under 30,000 or 50,000 gates, you could do it as ASIC. Once you got over 50,000 gates, it had to be a full custom chip.
BY MR. TETER:
Q. Do you remember which kind the Viper was?
A. That was a gate array.

65

Q. So the Viper would have been one that would have had a nine-month development cycle?
A. Correct.
MR. GRACEY: Object to form.

249. Based on OPTi's interrogatory response, the Compaq inventors, including Mr. Collins, filed their patent application before OPTi reduced its presnoop invention to practice. Filing the Compaq patent application was a "constructive reduction to practice" of the Compaq invention, according to the law as stated in the jury instructions that I cite above.

250. In addition, the testimony shows that the Compaq inventors reduced their invention to practice no later than February 1994, when they received the silicon chips for the CMC/DMC chipset and confirmed that those chipset worked for the PCI specification commands (which would necessarily include MRM), using the PCIFACTs test software. In addition, the silicon was completely prescribed several months earlier, when it was taped out

251. As set forth below, the Compaq inventors also conceived their invention prior to Mr. Ghosh, and diligently worked to reduce that invention to practice. The

Compaq prior invention, and the work reducing it to practice, occurred at Compaq's Houston Texas facilities.

1. Compaq was a member of the PCI special interest group (PCI SIG).

252. As discussed above in the description of Intel's prior invention, Compaq was a member of the PCI special interest group, or PCI SIG. Mr. Collins testified:

101

Q Was Compaq part of the PCI special interest group?
A Yes, no doubt.
Q Was Compaq involved at the very beginning of the formation of the PCI special interest group? 4:57:33PM
A I don't know. I think that the initial idea of this type of connection interface bus was probably raised by Intel that Compaq then quickly joined. I do know that we had one of our lead architects as the PCI special interest group or SIG group, either the president or chairperson. So we were very active in this spec. 4:57:59PM
Q Would you say that Compaq was on the forefront of PCI design when it originally was introduced? 4:58:17PM
A Yeah, no doubt, no doubt, I agree.

253. As OPTi founder Jaswa acknowledged, because Compaq was a member of the PCI SIG, unlike OPTi, Compaq was a generation ahead of OPTi and other chipset makers. Compaq, like Intel, was able to work on PCI chipsets before OPTi.

2. The CMC/DMC architecture included a read prefetch buffer that could hold up to two cache lines in the bridge, allowing the CMC/DMC to burst across cache line boundaries without waiting for snoops at cache line boundaries.

254. The Compaq system was designed to increase performance by having a great deal of "concurrency." In other words, for example, the PCI master could read data from main memory while the processor was working on data in the cache. Mr. Collins testified:

12

Q What was the motivation for creating the triflex architecture? 1:53:00PM

13

A Performance. 1:53:02PM

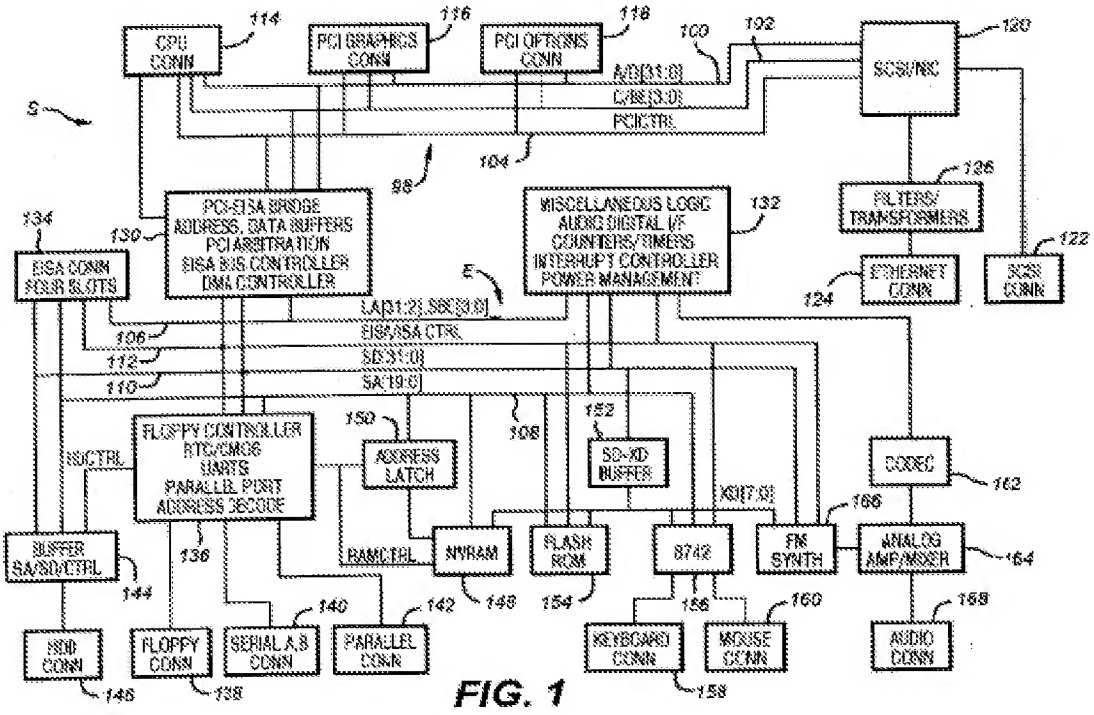
Q What specifically do you mean by performance?

A So overall system performance, so for example, the reason you have cache controllers on a processor is to keep the execution stream going while you're doing some background tasks. So in this case, this has a PCI bus that can read or write the local memory. This has a processor that can operate out of the local cache. And then this also has the connection from the processor to the PCI bus directly. 1:53:12PM 1:53:43PM

So in order to increase performance, the triflex is centered around concurrency, the processor itself can be running locally out of the cache and the PCI bus masters can be running locally out of the memory concurrently. 1:54:04PM

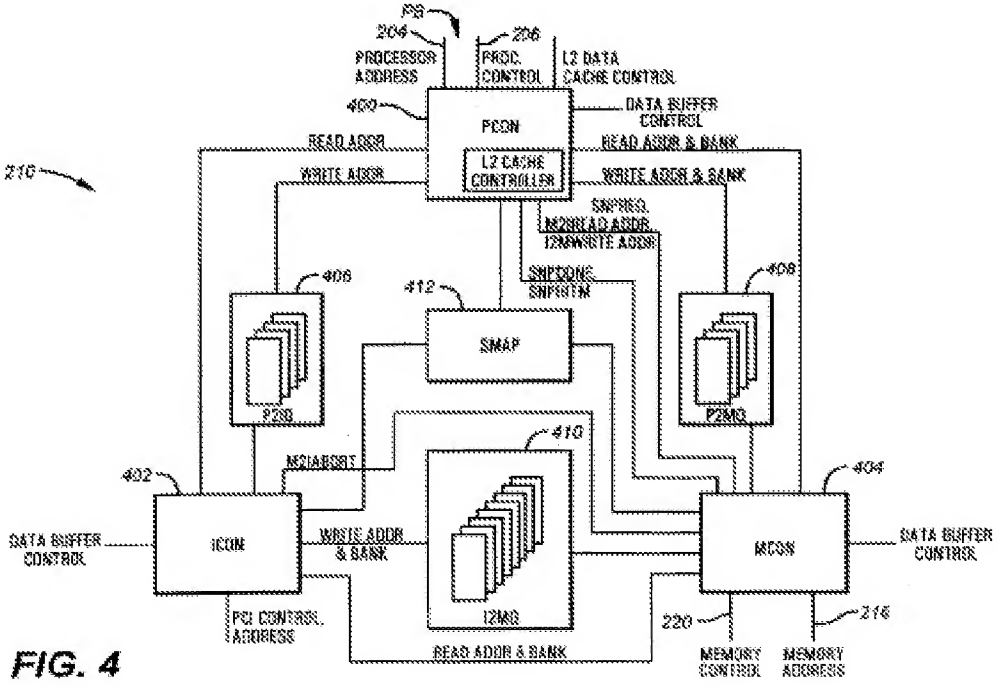
The main performance factor is the concurrency, so you can almost be doing two or three items at one time rather than one waiting for the other to finish. 1:54:28PM

255. The Compaq patent illustrated the system in Fig. 1.



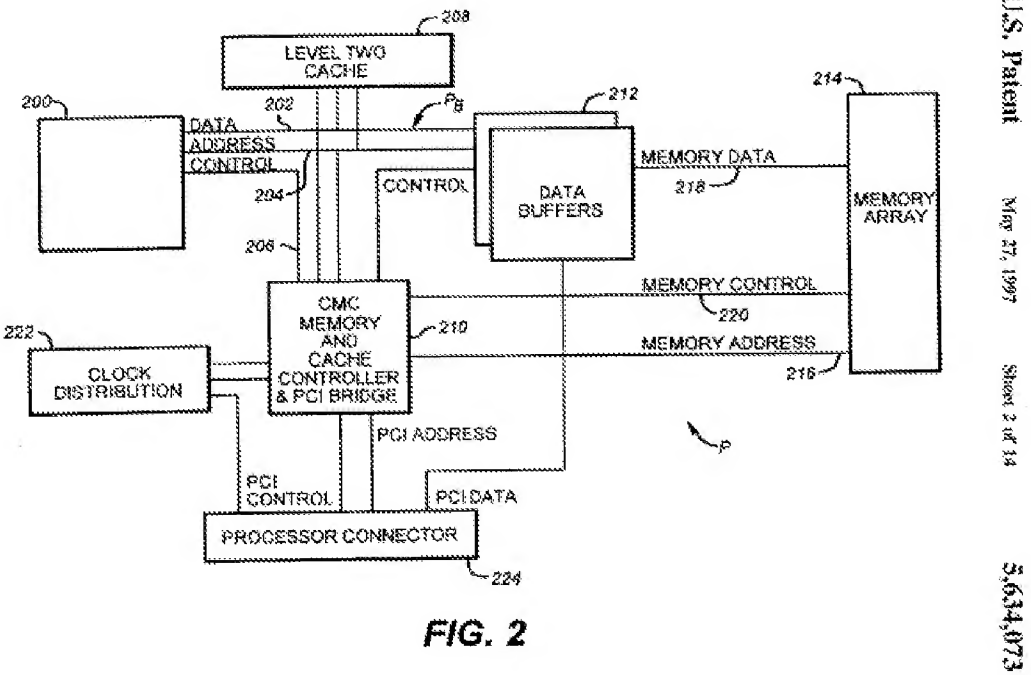
[EX. 182, '073 Patent].

256. Figure 4 in the Compaq patent shows the system's North Bridge.



[EX. 182, '073 Patent].

257. Figure 2 of the Compaq patent showed two squares, or two chips, for the data buffers (212).



[EX. 182, '073 Patent].

258. Mr. Collins, on behalf of Hewlett-Packard/Compaq as a designated corporate representative, testified that each square in item 212 of Fig. 2 corresponded to Fig. 5:

48

Q You referenced earlier with respect to Figure 2 that the data buffers, item 212 had two squares in its illustration?
A Yes. 2:53:56PM
Q I want to clarify what you meant by the two squares and whether Figure 5 represented a single square in item 212?
A On Figure 2 the CMC memory cache controller was a single chip which would be the 2:54:15PM north bridge. The 212 were two separate chips; to be frank it was called the DMCs. Those were two separate DMCs controlled by the CMC. And each one of those chips it was a structure of page 5. So it appears that page 5 is only showing the data path 2:54:43PM through one of the DMCs.
MR. BRODY: Figure 5?
THE WITNESS: Figure 5. So you can see just by going to the upper left, the PD, that should be 63 through zero. But in this case, it's 63 2:55:02PM

49

through 48, which is 16 bits. And then 31 through 2:55:07PM 16 which is another 16 bits. And there's a specific reason why that is 16 and 16 are 32, so you would have two chips. What's missing here is 47 through 32 and 15 through zero, there being the other chip. 2:55:32PM
BY MR. CUNNINGHAM:
Q Were there two item 414s, the PCI prefetch buffers?
A One in each chip.

259. Fig. 5 shows one of two chips in the Compaq chipset:

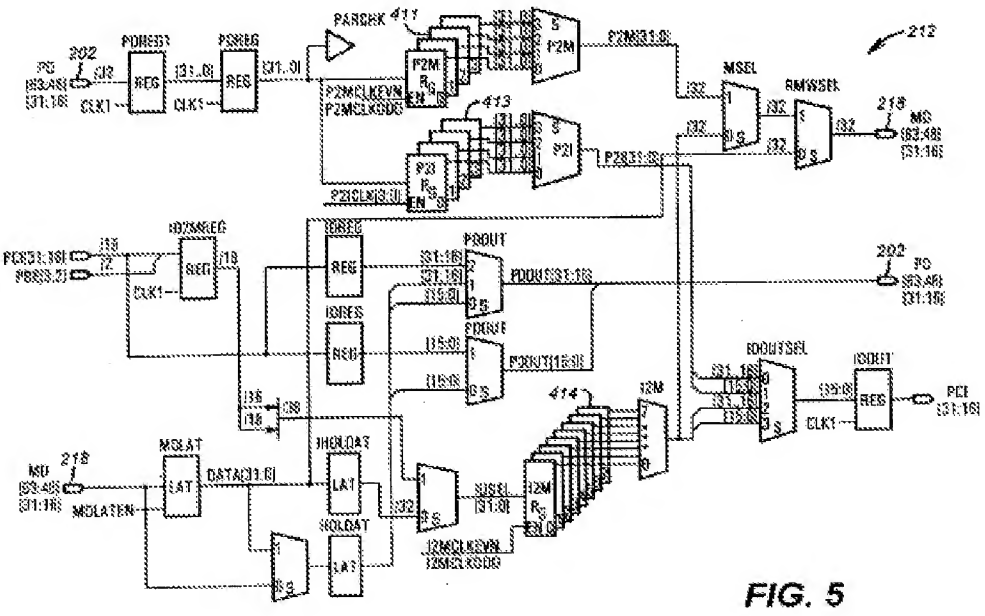


FIG. 5

[EX. 182, '073 Patent].

3. The CMC/DMC, as described in the Thome/Collins patent, performed a "pipelining" process that both presnooped and prefetched.

260. The Compaq patent describes a chipset for use in systems that supported the PCI bus. The system described in the Compaq patent was designed to work with both the 486 and the Pentium processor. The patent states:

> Referring now to FIG. 2, the processor board P is shown. In the processor board P of FIG. 2, the CPU or processor 200 can be any of a plurality of processors, such as the 486DX/33, 486DX2/66, 486DX4/50-100, 486DX4/33-100, 486DX4/33-83, P24T, PENTIUM 50/75, PENTIUM 60/90, and PENTIUM 66/100, and other similar and compatible processors.

[EX. 182, '073 Patent Col. 6:51-57].

261. Mr. Thome testified that the CMC/DMC was designed to work with a Pentium processor, and was the first chipset designed to optimize the performance of second generation Pentium processors.

12

Q. The first sentence of the first paragraph in Exhibit 184 reads, "TriFlex/PCI is the first system architecture designed to optimize the performance of PCI-based systems with second-generation Pentium processors." Do you see that?

A. Yes, I do.

Q. What were second-generation Pentium processors?
A. The first-generation Pentium processor was based on -- it was called the Pentium. It ran at 1666 [sic, 66] megahertz.
And the second one was a -- it was a cost reduction, it used a different silicone technology, and was able to run it faster, also, so -- up to 90 megahertz and then beyond. The internal code name for Intel is called P54C; whereas, the original Pentium was called P5.
Q. You said "P54C"?
A. Yes.
Q. Was the TriFlex/PCI the first system architecture designed to optimize the performance of PCI-based systems with the P54C Pentium processor?

13

A. To the best of my recollection, yes..

262. Mr. Collins also testified that the CMC/DMC was designed to work with a Pentium processor.

26

Q And of course the triflex PCI architecture was designed to be compatible with the Pentium processors? 2:15:31PM
A The answer is more than that. Is it all right? It's designed to work with any processor that uses a 64-bit data bus with a bursting of 4, which Intel followed that scheme but others did too. 2:16:03PM

263. Because the CMC/DMC was designed to work with the Pentium, it needed to work with a write-back cache. As a result, the CMC/DMC needed to snoop each cache line copied from secondary memory. The patent states:

> Snooping of read addresses as well as write addresses is necessary because the L2 cache controller, and the L1 cache controller in the processor 200 in certain cases, are preferably organized as writeback cache controllers, and therefore, snoop operations must occur on reads to maintain memory coherency.

[EX. 182, '073 Patent Col. 8:47-52].

264. Mr. Thome testified:

17

Q. (BY MR. TETER) Did the CMC/DMC perform snoops for read accesses by a PCI master?
A. Yes, it did.
Q. Can you explain that?
A. If a PCI master attempts to read some data location from main memory, then the CMC would -- at the

same time that it was reading main memory, would also provide a snoop operation to the second level cache that was controlled by the CMC, as well as the first level cache that was integrated in the P54C processor chip.

And so we would provide a snoop operation, inspect it, see if it had -- had the same data [as] main memory, if it was -- then you can leave it alone, or if it had different data, then we would have the cache write back the -- the modified data back to the main memory. And then we would provide that data back to the I/O master on the PCI bus.

265. The patent required each read to be snooped. The patent explained:

> Snooping of read addresses as well as write addresses is necessary because the L2 cache controller, and the L1 cache controller in the processor 200 in certain cases, are preferably organized as writeback cache controllers, and therefore, snoop operations must occur on reads to maintain memory coherency. The PCON block 400 provides the SNPDONE and SNPHITM or snoop done and snoop hit to modified signals to the MCON block 404 to allow the MCON block 404 to proceed with the read or write operations or retry a read operation if appropriate.

[EX. 182, '073 Patent Col 8:47-56].

266. As Mr. Collins testified, every cache line that was transferred to the read prefetch buffer had to be snooped:

52

Q In the read ahead implemented by the triflex PCI architecture did every cache line that came into that PCI read prefetch buffer get snooped? 3:00:54PM

A Yes.

267. Because CMC/DMC was designed to work with the PCI bus, it needed to respond to the "memory read multiple" or MRM command. Mr. Thome testified about his understanding of the MRM command as follows:

26

Q. First of all, what's a Memory Read Multiple command, do you know?

A. So the PCI bus specification had two versions of -- of memory reads. The Memory Read Multiple, it was kind of a hint to the memory controller to say, "I'm" -- "in addition to making this first request, I'm going to keep on requesting more data in a sequential manner." So the very next address and the address after that.

268. The Compaq patent described a chipset that used prefetch buffers to perform a memory read multiple, or MRM, command. The patent explains:

> The PCI bus provides opportunities to increase overall system performance, particularly that of the memory system. One of the read operations defined for the PCI bus is what is termed as a Memory Read Multiple cycle, which is used to indicate a desire to read a number of cache or memory lines, not just a single line. As noted in the PCI bus cycle definition, this cycle decoding provides an opportunity for the memory controller to start doing read aheads or pipelining so that the data can be obtained prior to actually being required on the PCI bus. With this data then obtained, the memory controller can allow access by the processor, thus further increasing overall system concurrency. However, it is also common for PCI bus masters to abort cycles prior to their completion, and if such an aborted cycle were to occur shortly after a Memory Read Multiple cycle has commenced, then a read ahead operation would have been started and would conventionally complete, only to have the data then immediately discarded. This would reduce overall system performance because of the wasted operations needed to start and complete the full read ahead operation. Therefore, it would be desirable to reduce the wasted time when doing read aheads during Memory Read Multiple cycles when the cycle is aborted early by the bus master on the PCI bus.

[EX. 182, '073 Patent Col. 2:40-63].

269. The Compaq chipset, the CMC, was designed to both prefetch and to presnoop in PCI burst read transfers. The Compaq chipset had a read prefetch buffer, called the "I2M queue", that could hold two cache lines of data. The I2M queue is shown in Figure 5, above. Mr. Collins testified about the I2M prefetch buffer as follows:

46

Q Do you understand what I'm referring to when I use the term "prefetch buffer"? 2:49:39PM
A Yes.
Q What is your understanding of that term?
A The prefetch buffer is a buffer in the design that, it's sort of a speculative read, for example. So based on the PCI spec, if the PCI spec is issuing a read of memory that asks for read multiple, in some way the second cache line is speculative. You don't know if the master, the PCI master really needs it or not. So it's a way of actually doing bursting. 2:49:53PM 2:50:30PM
Q Did the triflex PCI architecture as described in the '073 patent implement a PCI read prefetch?
A Yes. So you actually asked me earlier, so Figure 5, so the mechanics are that this is a 64-bit memory data bus that muxes itself down to a 32-bit PCI bus. And it takes two data buffers to do that. 2:50:44PM
This I2M Queue which is number 414, if you look at that, you can see that it's 7 queue 2:51:20PM

47

locations deep. So earlier -- 2:51:25PM
Q Perhaps 8 if it goes to zero to 7?
A Yes, sorry, 8 deep. So you asked earlier a cache line is four times the 64 bits. This buffer being 8 is two times the cache line. So what this 2:51:48PM does is when the initial read of the bus master, the PCI bus master starts, it will load up the first four locations of this buffer. And if it's a read multiple, then when that data is transferred we can transfer the next cache line into this buffer. 2:52:19PM That's why it's actually a depth of 8.
The structure is such a way that once you start bursting out the first cache line you can begin filling that buffer with the next cache line, which is a pipelining aspect. 2:52:37PM
Q So item 414 in Figure 5 is the structure that would be the PCI read prefetch buffer?
A Yes.
Q Would you mind circling that on your copy and labeling it as a PCI read prefetch buffer? 2:52:52PM
A I should do that?
Q Yes, please, so it's clear for the record.
A You want a PCI read prefetch.

270. Mr. Collins testified that the prefetch buffers in the system were capable of storing two cache lines:

70

Q How many lines can be stored in the I2M Queue?
A Two. That's why it's eight locations deep.

71

Q So in every implementation of the triflex PCI architecture the PCI read prefetch buffer would be capable of storing two cache lines?
A Yes.

271. Mr. Thome also testified that the CMC/DMC buffers could hold two cache lines of data.

25

Q. In the portion of the description in the White Paper we were just looking at, it indicated that there is a 64-byte buffer. How many cache lines could be stored in the read buffer in a CMC/DMC?
A. Two cache lines.

272. The presence of buffers able to hold more than one cache line supports the testimony of Mr. Thome and Mr. Collins that the Compaq CMC/DMC both presnooped and prefetched. Having space for two cache lines of data in the I2M queue allowed the CMC/DMC to stream across cache line boundaries without waiting for a snoop.

273. Having space for two cache lines, but not using them for presnooping and prefetching, would not make sense. As Mr. Collins testified, the "whole reason" for having prefetch buffers that were two lines long was to allow a PCI read to burst across cache line boundaries.

86

Q So the results with respect to the presnooping and prefetching is the that PCI bus master would receive a constant rate of data?
A Yes. Well, every clock there will be a data transfer, yes. 4:23:37PM
Q Will there with a data transfer across cache line boundaries?
A Yes, of course because this is two cache lines, so that's the whole reason the queue is 8.

4. The CMC/DMC, as reduced to practice in silicon and as described in the Thome/Collins patent, performed a "pipelining" process that both presnooped and prefetched.

274. Compaq released its CMC/DMC chipset in November 1994.

Q1. What Compaq computers currently utilize TriFlex/PCI?

A1. TriFlex/PCI was first introduced on the ProSignia 500 computer in November 1994. It has since been incorporated into the Deskpro XL Series and the ProLiant 1500 server. It was also incorporated into the new Deskpro and ProLinea Families introduced on March 8, 1995.

[EX. 183 at HP00256 (A Comparison of Compaq Triflex/PCI Architecture to the Intel Triton Chipset at 6)].

275. OPTi has questioned whether the CMC/DMC was able to perform presnooping when it was released in production in November 1994. Mr. Thome, on behalf of Hewlett-Packard/Compaq as a designated corporate representative, testified that Compaq released the CMC/DMC chipset in November 1994:

10

Q. Do you remember when the CMC/DMC chipset was first used in a commercial product?
A. We first started selling it in November of 1994.

37

Q. And you introduced the CMC/DMC chipset in November?
A. November of 1994.

39

Q. And underneath it, it reads, "TriFlex/PCI was first introduced on the ProSignia 500 computer in November 1994." Do you see that?
A. Yes.
Q. Was TriFlex/PCI introduced in the ProSignia 500 computer in November 1994?
A. So I recall it being somewhere in that time frame. This document seemed to indicate it was November 1994.
Q. And when we say "TriFlex/PCI," that TriFlex/PCI included the CMC/DMC chipset that we've discussed

40

earlier?
A. Yes.

136

TriFlex/PCI was first released to the public on what date?
MR. BRODY: Object to the --
A. TriFlex/PCI was implemented in computers at -- based on the -- the information in the document, it looks like it was November of 1994. And I recall it being in 1994, as well.

276. Mr. Thome testified that the CMC/DMC chipsets in the Compaq products released between November 1994 and March 1994 were based on the same chipset design, the CMC-2.

113

Q. This refers to the products containing the TriFlex chipset, and it gives -- lists three products, I think: the ProSignia product, which launched in November '94, the DeskPro product, DeskPro XL series, and the ProLiant 1500 server, which apparently launched subsequently. And it says it was also incorporated into the DeskPro and ProLinea families, introduced on March 8th of 1995, so that, generally, it's a HP 256?

A. Uh-huh. I see that.
Q. Do you know whether there were any changes in the chipset between those product launches?
A. No, there wouldn't have been any changes in the chipset. They were all based on the CMC-2 chipset, that I can recall.

277. Mr. Thome testified that the CMC/DMC chipset was thoroughly tested and debugged and fully functional:

137

Q. (BY MR. TETER) Does plaintiff's Exhibit 103 indicate to you that Compaq released the TriFlex/PCI to the public in November 1994, without having done the necessary testing?
MR. BRODY: Object to the form.
A. Compaq would have not -- Compaq would not have released the product until -- until we thought it was thoroughly tested and debugged and fully functional.
THE REPORTER: Debugged and?
THE WITNESS: Tested and fully functional.
MR. DAWSON: Fully functional.
Q. (BY MR. TETER) When Compaq released TriFlex/PCI and the CMC/DMC in November 1994, did Compaq have any opinion about whether it was fully debugged and fully functional?
MR. BRODY: Object to the form.
A. We believe that it was fully debugged and fully functional.

278. Mr. Thome testified that the CMC/DMC chipset had a fully functional MRM command as of November 1994:

141

Q. Would you have shipped the CMC/DMC chipset without satisfying yourself that the Memory Read Multiple command would function as you contemplated?
A. I don't recall thinking that we would do something like that because without the Memory Read Multiple command operating, we wouldn't be able to run the PCI bus at the full bursting speed that we talked about.
Q. And so you wouldn't have shipped product with that feature in it until you had satisfied yourself that it was fully functional and operational, right?
A. I think we would have made sure that it was functional before we would ship that product.
Q. Okay. Thank you.
MR. BRODY: That's all I have.

142

FURTHER EXAMINATION
BY MR. TETER:
Q. And one last question.
There's nothing here today that leads you to believe that it -- that feature wasn't fully functional as of November 1994; is that right?
MR. BRODY: Object to the form.
A. So to the best of my recollection, that feature was functioning when we shipped the product.

279. OPTi counsel asked whether it was possible that the CMC/DMC was shipped with the MRM command turned off in November 1994. Mr. Thome testified that he did not believe that was true.

142

Q. Is it possible that in November of '94, you shipped product with that feature turned off and that you enabled it later in March when you shipped -- in March of '95, when you shipped further product?
MR. TETER: Object to form.
A. So to my recollection, we didn't respin the silicone right then. We would have to -- you know, if we could find documentation showing whether we did or not, but I don't recall respinning it, meaning doing another version of the silicone right then.
If we -- you know, so -- so if we were

143

using the same silicone all the way through, we wouldn't have had the ability to correct any bugs that existed, you know, from one time to the next.

280. Mr. Collins also testified that the CMC/DMC prefetched and presnooped in response to an MRM command, which is what allowed the system to transfer data on every clock, or achieve the full data rate possible with the PCI bus (with the caveat that another cycle, by the processor for example, did not take priority):

86

Q In normal operation would the triflex PCI architecture presnoop in response to a memory read multiple command? 4:22:44PM
A It will try.
Q And what effect would this presnooping and prefetching have on the data flow from the perspective of the PCI bus master?
A It will be bursting data. It won't be 4:22:58PM

interrupted. So, for example, on the PCI bus, every PCI clock there can be a data transfer transaction. If both the master and the slave are enabled with the readies, you can move data with every clock. It would be the goal of the triflex chip to move data every clock.

Q So the results with respect to the presnooping and prefetching is the that PCI bus master would receive a constant rate of data?

A Yes. Well, every clock there will be a data transfer, yes.

Q Will there with a data transfer across cache line boundaries?

A Yes, of course because this is two cache lines, so that's the whole reason the queue is 8. 4:23:23PM 4:23:37PM 4:23:53PM

87

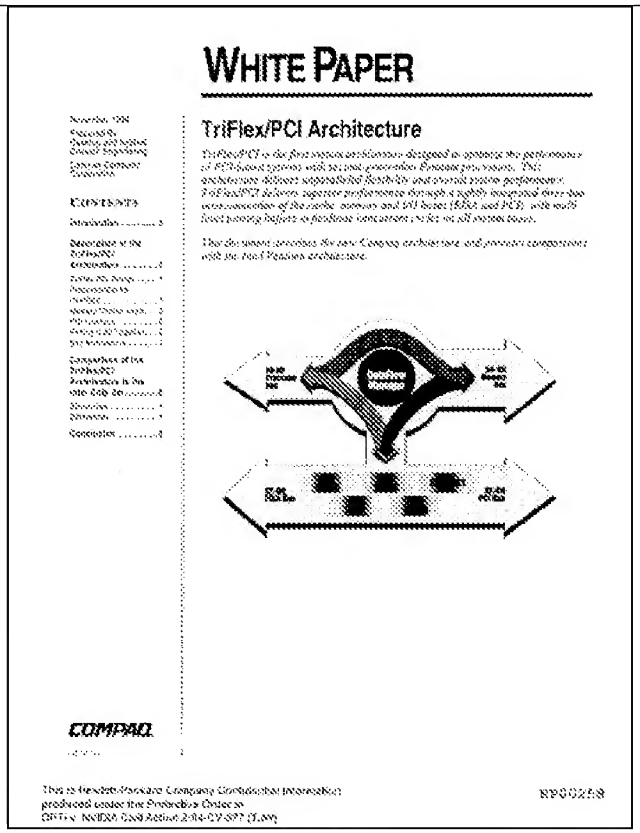
Q So as long as the PCI bus master is requesting data the presnoop and prefetch will allow that data to be fed to the PCI bus master continuously? 4:23:56PM

A Yes. Well, that is the goal. 4:24:07PM

Q In normal operation the PCI bus master will receive the data continuously?

A Not quite. In normal operation, if there was no other activity asking the memory to do some other work, if the memory was idle and there was constantly read requests coming up or a read multiple -- I don't remember the priority -- but if the PCI read is not conflicting with priority of the memory, then the memory is available, and it will continually do rereads and snoops and keep the bursting going of the PCI master, yes. 4:24:26PM 4:24:52PM

281. The written evidence, discussed below, supports the testimony of Mr. Collins and Mr. Thome that Compaq had working silicon in November 1994 that had the CMC/DMC capability. Compaq published a whitepaper describing the CMC/DMC during November 1994. Because Intel had not yet released the Triton chipset, Compaq compared the Triflex/PCI, and its CMC/DMC bridge, with the Intel Neptune chipset.



WHITE PAPER

TriFlex/PCI Architecture

COMPAQ

[EX. 184].

282. The operation of the Compaq CMC/DMC chipset is described both in the whitepaper, (EX. 184), and in the Thome/Collins patent, (EX. 182).

283. Mr. Thome, on behalf of Hewlett-Packard/Compaq as a designated corporate representative, testified that the CMC/DMC chipset released in November 1994 operated as described in the patent.

80

Q. Is it your understanding that the process for handling Memory Read Multiple commands in the CMC/DMC silicone that you received in February 1994 is the same process that you then patented in the 073 patent, Exhibit 182?
MR. BRODY: Objection; form. Calls for --
Q. (BY MR. TETER) Do you understand the question?
MR. BRODY: -- a legal conclusion.

81

A. So -- so the way we -- so from CMC-1 to CMC-2, we didn't make any substantial functional changes. Nearly everything we -- we changed were to fix bugs that we found, either in our -- our simulations or in our testing in the lab.

And so -- so any changes would have not been from intent in what the design was to do, but because the design, as we developed it in the CMC-1, had some things that it didn't do what we intended to do, such as the errata that we talked about just a few minutes earlier.

81

Was it your intent to describe, in Exhibit 182, the CMC/DMC chipset that we've been discussing here

82

today?

MR. BRODY: Object to the form.

A. So, yeah. So as -- as I can -- as best I can recall, the intent was for this patent to be a reflection of some aspects of the design of the CMC/DMC.

Q. (BY MR. TETER) And focusing on the way the CMC/DMC handled a Memory Read Multiple command, are you aware of or can you recall any differences between the CMC/DMC silicone, that you received in February 1994, and the system described in the patent, Exhibit 182?

MR. BRODY: Object to the form. Asked and answered.

A. So -- so as -- so as we defined and designed it in February of 1994, the CMC-1 that we received, is essentially the same as what we patented. And any differences would have been, you know, errata for the handful of bugs that we found along the -- along the line.

284. The whitepaper described the CMC/DMC PCI interface as follows:

> **PCI Interface**
> TriFlex/PCI is optimized for both processor accesses to the I/O bus and for bus master (EISA and PCI) accesses to main memory on a PCI bus running at 33 MHz.
>
> TriFlex/PCI can send data at 133 MB/s to devices, such as a graphics controller, using the PCI burst protocol.
>
> PCI bus masters may access system memory at the maximum PCI transfer rate, or 32 bits every PCI clock (133 MB/s), improving performance of devices such as the integrated 32-bit Fast SCSI-2 and NetFlex-L ENET controller. Writes from bus masters to system memory are posted in a 64-byte write posting buffer. Reads from bus masters are also stored in the 64-byte buffer, allowing the bus master to read additional data at the 133 MB/s transfer rate. These buffers ensure the bus master maintains the 133 MB/s transfer rate up to 94 percent of all data transfers.
>
> TriFlex/PCI allows EISA and PCI bus masters to access system memory while simultaneously queuing up processor accesses to the I/O bus. This ensures a minimum stall time for the processor and the I/O bus.

[EX. 184 at HP00263].

285. In particular, the Compaq white paper noted that a bus master could read "additional data at the 133 MB/s transfer rate" and stated that the read prefetch buffers

"ensure the bus master maintains the 133 MB/s transfer rate up to 94 percent of all data transfers."

286. Compaq's statements about the performance of the CMC/DMC bridge support Mr. Thome's and Mr. Collins' testimony that the CMC/DMC chipset used both prefetching and presnooping. CMC/DMC was using prefetching to eliminate DRAM latency between the memory and the bridge, and using presnooping to eliminate the latency that otherwise would occur waiting for a snoop. Thus, the prefetching and presnooping allowed the CMC/DMC to achieve the transfer rates that Compaq recited in the whitepaper.

287. Because the Compaq system both presnooped and prefetched, it allowed the PCI master to burst at the fastest possible rate across the PCI bus. Mr. Collins testified:

88

Q So the triflex PCI architecture when implementing the presnoop and the prefetch allows the PCI bus master to burst at the fastest possible rate? 4:25:47PM
A Yes, yes. That's the goal.

[redacted]

289. Mr. Collins testified that the Compaq chipset could burst across cache line boundaries as follows:

93

Q The result of the triflex PCI architecture doing presnooping and prefetching, the PCI bus master can cross cache line boundaries, correct? 4:33:00PM
A Yes.
Q Can burst across cache line boundaries? 4:33:11PM
A Yes.

290. Recall that Intel, when it was designing Triton, also recognized that to achieve maximum PCI throughput, it should both presnoop and prefetch. If Compaq's Triflex/PCI bridge prefetched but did not presnoop, it would have had a lower throughput for

PCI burst reads. If Compaq's Triflex/PCI bridge presnooped but did not prefetch, it would have had a lower throughput for PCI burst reads.

291. If either the prefetching or presnooping features of the CMC/DMC chipset was malfunctioning in November 1994, Compaq would not have been able to make the performance claims that it included in the whitepaper. Compaq's performance claims, set forth in the whitepaper, support Mr. Thome's and Mr. Collins' testimony that the CMC/DMC released in 1994 performed both prefetching and presnooping.

292. Comparing the Triflex/PCI bridge with the Neptune, the Compaq white paper noted that the Compaq CMC/DMC bridge achieved a sustained throughput of approximately 125Mbytes/second, compared with Neptune's approximately 75 Mbytes/second throughput.

Differences

	Compaq Triflex/PCI	Intel Chip Set
Numbers based on:	100/66 MHz Processor	100/66 MHz Processor
DataFlow Manager	Y	N*
Processor to I/O buffer (bits)	256	128
I/O to memory buffer (bits)	512	256
Cache	2-way	direct map
Pipelining supported	Y	Y**
Pipelined memory bus	Y	N
Max. number of cycles simultaneously	16	8
Max. sustained process/memory bus bandwidth (MB/s)	133	92
Sustained I/O bandwidth (MB/s)	123	75

[EX. 184 at HP00265].

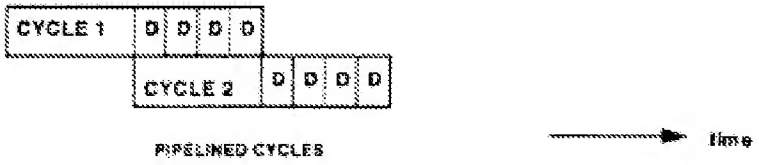
293. The Compaq white paper emphasized that Triflex/PCI performed pipelining even when a PCI bus master was reading from secondary memory, unlike the Intel Neptune chipset. The Compaq white paper explained:

> TriFlex/PCI is designed with pipelining on the processor bus with EISA and PCI bus master concurrency. TriFlex/PCI always uses pipelining to improve processor performance, even when handling bus master accesses. This is unlike the Intel chip set, which does not allow pipelining when a bus master is running. Further, on the Intel chip set, many pipelined cycles run at the same speed as a non-pipelined cycle. The pipelining support of TriFlex/PCI results in significantly higher sustainable processor/memory and I/O bus bandwidths. Because of the concurrency between the memory pipelining and I/O bus, the memory bus is not left idle when there are cycles that may be run.

[EX. 184 at HP00266].

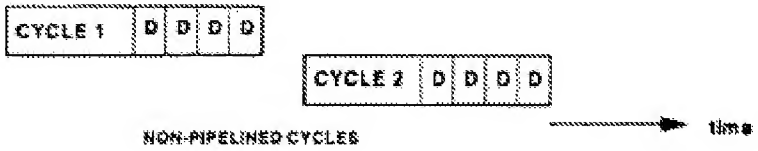
294. Again, the difference in performance between the CMC/DMC chipset and the Intel Neptune supports Mr. Thome's and Mr. Collins' testimony that the CMC/DMC performed both prefetching and presnooping.

295. The whitepaper contains a description of a process called "pipelining" that illustrates how the presnooping and prefetching worked in the CMC/DMC chipset. The "pipelining" described in the Compaq white paper was a process where one line starts before the data transfer of the previous line is complete. The Compaq white paper illustrated the pipelining process as follows:



[EX. 184 at HP00262].

296. The Compaq white paper contrasted the pipelining performed by the Compaq/ Triflex/PCI chipset with a non-pipelined system, such as Intel's Neptune chipset:



[EX. 184 at HP00262].

297. Mr. Thome, on behalf of Hewlett-Packard/Compaq as a designated corporate representative, testified about the CMC/DMC "pipelining"

Q. And the second paragraph states, "TriFlex/PCI uses a pipelined memory bus to sustain the maximum throughput of 133 MB/s from the memory. The following diagram illustrates the benefits of pipelining." Do you see that?
A. Yes.
Q. And then there are two diagrams beneath: one for nonpipeline cycles and one more pipeline cycles?
A. Yes.
Q. Do you see those?
A. Yes.
Q. Can you explain what pipelining means, in the context of TriFlex/PCI?
A. So pipelining is a -- is a technique to -- to get data to -- to come out faster. So in a typical -- the typical way you access memory is, you first present an address to the memory controller and say, "Could you go, please, get the data that's located at this address."
And then once -- then the memory controller will go get the data, which takes a certain amount of

19

time, and then return the data back.
And so if you do that sequentially, then what happens is, you have to wait until the data returns before you can say, "Okay, now here's another address."
And since there's a -- as you can see in the picture there, there's an amount of time between when you present the address before you start getting the data returned. You have to wait that time again before you can -- you can go to another cycle.
With pipelining, what we do is, we present the address and then the memory controller starts getting the data. And before the memory controller has even returned that first data, we present another address, and say, "Okay, go ahead and start on this one right behind it."
And so what that allows us to do is, as soon as the memory controller has returned the first data, it already is -- is working on getting the second data without waiting. And so the net result of that is that we can return data to the requestor at a -- at a much faster pace because we're not waiting for the data to be returned before we start asking for the next address.

298. In the context of a PCI burst read, the pipelining would work as follows. The first line would be fetched by the I2M prefetch queue on the CMC/DMC bridge,

and while the first line was being transferred to the bus master, the second line would be snooped and transferred to the I2M prefetch queue. When the first line had been received by the bus master, the second line would begin to transfer to the bus master without delay for the snoop. At that time, while the second line was being transferred to the bus master, the third line would be snooped, and would then begin to transfer to the buffer. The Compaq pipelining process, would thus use both presnooping and prefetching and would allow the bus master to read data from memory at or very near the maximum rate allowed by the PCI bus.

299. Mr. Thome described the process in the context of a PCI burst read as follows:

25

Q. Did the CMC/DMC have to stop at cache line boundaries to snoop?
A. So we would have to snoop each cache line.
Q. Did that slow down the 133 megabyte per second transfer rate of the CMC/DMC?
A. Only if the snoop resulted in what's called a snoop hit, where the -- the cache line was modified, data verses what was in main memory.
Q. Let's assume that there's a snoop miss, okay?
A. Okay.
Q. In the case of a snoop miss, would the need to do the snoop slow down the data transfer to a PCI bus master?
A. No.
Q. Can you explain why not?

26

A. We would -- once we got a -- a request from a PCI master, we would issue the snoop request to the processor at the same time we would issue the main memory request. And so we could do -- we could almost always snoop the processor faster than the read cycle would take from main memory. And so the snoop would be complete before the memory cycle was complete.
Q. If we could sort of walk through one example. So let's say a PCI bus master requests Memory Read Multiple command starting at Line 1, okay?
A. Okay.
Q. First of all, what's a Memory Read Multiple command, do you know?
A. So the PCI bus specification had two versions

of -- of memory reads. The Memory Read Multiple, it was kind of a hint to the memory controller to say, "I'm" -- "in addition to making this first request, I'm going to keep on requesting more data in a sequential manner." So the very next address and the address after that.

Q. So in our example, we'll have a Memory Read Multiple command starting at Line 1, okay?

A. Okay.

Q. And can you explain the process of requesting data from main memory, transferring data to the PCI master, and snooping, with a Memory Read Multiple

27

command that starts on Line 1?

A. Okay. So when the -- when the PCI bus issues a Memory Read Multiple request, we would then issue a memory read command to the -- the memory controller, and at the same time, issue a snoop request to the cache controller.

Q. Snoop for what line?

A. For the same line.

So -- so Line 1 -- so we do show a request for Line 1, in the example, to the memory controller and a snoop request for Line 1 to the caches.

And then we would let them both run independently, so one would complete before the other. Typically, the snoop would complete first. And if the snoop said that there was no modified data in the caches, then we would just simply wait for the memory read to complete and allow the memory to get buffered to return to the PCI bus.

We would also, using the pipelining that we talked about earlier, issue a second memory read command, before the first memory read command completed, of Line 2, to the memory controller, such that we could then continue to -- to read data from the -- from the memory at the faster possible pace.

And then when we issued the second read

28

command, we would also, at the same time, issue a second snoop command for Line 2, at the same time, as well.

And -- and everything would continue going that way, assuming everything the snoop misses, and you would just keep on going.

In the case of a snoop hit, we would wait for the memory read to complete from the cache line, because the way memory works, you can't really shut it off in the middle. You have to run it to completion.

And then -- then we would write back the data from the processor or from the second cache -- second level cache into main memory. Then we would discard the read that we just completed to the PCI bus

and issue a new read to reread the data that was discarded, and store that into the -- the buffer, and then present that back out to the PCI bus.
Q. In the case of a snoop miss scenario, that you were describing just a moment ago, would the snoop of the next line take place prior to the completion of the transfer to the PCI master of the current line?
A. It could.
Q. Can you explain the answer -- your answer?
A. Yes. Since the PCI master is running at a different speed from the main memory, there is no absolute guarantee or requirement that -- that the

29

sequence would happen necessarily one versus the other.
Q. In order to achieve the 133 megabyte transfer rate of burst reads to a PCI master, would the snoop of the next line have to occur before the current line has completed transferring all the way to the bus master?
A. To the best of my recollection, yes.
Q. And is that how it worked in normal operation?
A. So to the best of my recollection, yes.

300. Mr. Collins testified that the Compaq system would try to snoop ahead in normal operation:

85

Q Would the triflex PCI architecture snoop ahead in normal operation?
A It would try. There would be no doubt that if the bus master asked for a read multiple and the initial data only occupies half the I2M Queue and there is room for another line, and when that transaction is done, would we go and ask the memory again for the next line and snoop the next line with the other half of the I2M Queue empty, and the answer is yes. 4:21:27PM 4:21:50PM

301. Mr. Collins' testimony indicates that the trigger for the next line snoop would be when the I2M prefetch buffer had space for another line. Space in the I2M buffer would depend on the status of the transfer of the current line. In other words, the next line snoop would be triggered when the current line began to transfer to the PCI bus master, because that is when the buffer would have space available for the next cache line. The model instruction for anticipation in view of the prior art states:

> In deciding whether or not a single item of prior art anticipates a patent claim, you should consider that which is expressly stated or present in the

> item of prior art, and also that which is inherently present. Something is inherent in an item of prior art if it is always present in the prior art or always results from the practice of the prior art, and if a skilled person would understand that to be the case.

302. A skilled person would understand that the trigger for the CMC/DMC chipset would be based on the assertion of TRDY and IRDY at the beginning of the current line. Thus, the trigger for a next-line snoop would be the same as the trigger in the '906 patent – as soon as the current line is transferring on the PCI bus, the next line is snooped.

303. The description of the trigger in the '073 patent is consistent with Mr. Collins' testimony:

> The ICON block 402 receives this M2INA signal and again checks to see if there is space available in the I2M queue 414 to receive another cache line. When there is room because the ICON block 402 has provided sufficient data to the PCI bus 98, the next M2I read request is provided to the MCON block 404. This process continues until either the Memory Read Multiple completes, a page boundary is crossed or the PCI bus master aborts the cycle.

[EX. 182, '073 Patent Col. 22:29-37].

304. To determine whether space was available during a bursting operation, the CMC/DMC would have to consider the TRDY, FRAME, and IRDY signals on the PCI bus. If the TRDY, FRAME, and IRDY signals were asserted for the first data unit transfer of the line N, the CMC/DMC would know that one cache line in the I2M buffer had become available. In that case, the CMC/DMC trigger would conduct a snoop of line N+1.

305. The whitepaper's description of the CMC/DMC's operation would have been very misleading unless the CMC/DMC chipset had operated as Mr. Collins and Mr. Thome testified in November 1994. There is no reason to believe that Compaq, a publicly-traded company, and its Texas-based engineering team, would have purposely deceived customers by advertising a bursting capability and pipelining feature that did not work for MRM commands. With these features correctly implemented, an easily-verifiable PCI transfer rate could have been measured and compared to the theoretical maximum. Without prefetch/presnoop, maximum measured PCI throughput would have been far lower than the peak. Had their chipset not really implemented the claimed features, the deception would have

been quite obvious. The evidence is clear and convincing that the CMC/DMC chipset, as released in November 1994, implemented presnooping and prefetching, as Mr. Thome and Mr. Collins testified.

306. After Intel released the Triton chipset, Compaq emphasized the Triflex/PCI features. Intel's Triton was capable of bursting across cache line boundaries, but so was Compaq's Triflex/PCI, and Compaq argued that its chipset had a richer feature set. Compaq argued:

> **PCI to Memory Interface**
>
> Intel states that enhancements to their PCI to memory buffers enable bus master transfers from disk to memory to run much faster on Triton than on previous Neptune-based systems, however Triton's >100MB/s peak PCI to memory transfer rate still falls short of TriFlex/PCI. TriFlex/PCI is capable of a peak transfer rate from the PCI bus to system memory of 133MB/s, and capable of sustaining a transfer rate of 125.5MB/s. Further, Triton can only support their high PCI to memory transfer rate as long as the CPU is not accessing the second level cache or system memory. When the CPU accesses the second level cache or system memory in a Triton-based system (which is not an uncommon occurrence), PCI to memory bandwidth falls to zero. Conversely, when a PCI master accesses system memory on Triton, the CPU is prevented from further accesses to either system memory or the second level cache. In contrast, the sophisticated Data Flow Manager of TriFlex/PCI places no such constraints on either bus. With TriFlex/PCI, both PCI bus master and the processor may run simultaneously, largely unimpeded, allowing a higher overall system throughput.

[EX. 183 at HP00251].

5. The CMC/DMC chipset, as described in the Thome/Collins patent and as reduced to practice in silicon, did not disconnect at cache line boundaries, but limited bursts only to entire pages of memory and/or 256 addresses.

307. In ordinary operation, because the CMC/DMC allowed MRM transfers to burst across cache line boundaries without disconnecting, the transaction would continue until one of the following events occurred—either the MRM completed, a page boundary is crossed in main memory, or the PCI bus master aborts. [EX. 182, '073 Patent Col. 22:35-37 ("This process continues until either the Memory Read Multiple completes, a page boundary is crossed or the PCI bus master aborts the cycle.")].

308. The Compaq patent also described a counter that would limit the length of burst read transactions by a PCI master. The counter would allow a PCI master to burst across

many cache lines, but after long bursts of many cache lines, would require the master to relinquish the bus. The '073 patent states:

> This M2INA signal is an indication that the processing of the prior address is complete by the MCON block 404 and the processing of the next read address can begin. The actual incrementing of the read address value is performed in the ICON block 402 using an 8 bit counter, thus limiting the total read ahead length to 256 address values. When the counter reaches 255, the read ahead operation is terminated by logic not illustrated for simplicity by causing the Memory Read Multiple to be disconnected. A new address must be received from the PCI bus master to continue the Memory Read Multiple Operation.

[EX. 182, '073 Patent Col. 23:23-34].

309. Mr. Thome, testifying on behalf of HP/Compaq, testified that the counter allowed the MRM burst read to cross up to 255 cache line boundaries in a single burst.

31

Q. If you could look, still, at the patent on Column 23 -- Column 23, Line 26?
A. Okay.
Q. There's a description of, "The actual incrementing of the read address value is performed in the ICON block 402 using 8 bit counter, thus limiting the total read ahead length to 256 address values. When the counter reaches 255, the read ahead operation is terminated by logic not illustrated for simplicity by causing the Memory Read Multiple to be disconnected. A new address must be received from the PCI bus master to continue the Memory Read Multiple Operation."
Do you see that?
A. Yes, I do.
Q. Can you -- well, first of all, was that counter included in the CMC/DMC that was commercialized?
A. Yes.
Q. Can you explain how that counter worked?
A. So when -- when we receive a Read Multiple Command, it -- we were responsible for incrementing the

32

address for each additional cache line boundary. The PCI master would just simply give us the first command -- or first addresses and then we had to keep on adding new point -- new addresses.
So the counter was used to add -- to increment the address for each -- each additional request. Because it was an 8-bit counter, at some point in time, it would -- you would get to the end, and then at that point, you would have to terminate the PCI transaction and force the PCI master to issue a new --

new request with a new address.
Q. So how many cache lines could you cross in one Memory Read Multiple Operation, using the counter that you had in the CMC/DMC?
A. So to my recollection, it would be 256 total cache lines that -- you could cross 255 at the maximum.
Q. Why did you decide to include a counter?
A. If we did not include a counter, then we would have had to force the PCI master to issue a command -- a read command for every single cache line that it was requesting.

33

Why did you decide to have a 256 address value limit for your counter?
A. It seemed like a practical limit.
Q. Why did you decide to have any limit at all for

34

the counter?
A. For purposes of not getting too many gates. As -- as you increase the size of a counter, it -- it grows dramatically how many gates or transistors are required to implement the counter.
Q. Was it your understanding that the 255 cache line limit that you could handle in one MRM transaction was sufficient for most applications?
A. Our -- our belief was that 256 cache lines would be sufficient for performance for most applications.

310. Mr. Collins couldn't recall whether the counter kept track of cache lines or memory addresses, but in any event, testified that the counter allowed a MRM transfer to cross many cache line boundaries:

97

Q So when the patent refers to in column 23 a total read ahead length of 256 address values, is it saying that the total read ahead length is 256 cache lines? 4:51:37PM
A That's the part I couldn't find. I don't know if it's 256 cache lines or it's 256 PCI 4:51:55PM

98

addresses. From what I can tell reading the different areas of the patent is this is really 256 addresses based on a PCI bus. So this is 256-bit addresses of 32 bits. 4:52:02PM
Q Is it accurate to say that the ICON block is keeping track of 256 address values that it is presenting to the MCON block? 4:52:22PM

MR. BRODY: Object to the form.
THE WITNESS: Yes. It's important to note, though, that whatever address, whatever address is the initial address by a PCI bus master, that is really the base address. And that is the address that goes to memory to start to fill this queue. Then at every time that the data for that address is returned, you should be incrementing the counter.
So it's not like ICON has all these addresses, it's basically an incremental address. So for every data transfer you're adding one to the counter. So it's like a running total, 21, 22, 23, 24, it's an incremental thing up to, this case, 255 which is a full counter. So basically you can start anywhere with any address, and then every clock that you do a transfer will increment the counter one notch. And after 256 transfers, that counter will 4:52:39PM 4:53:08PM 4:53:27PM 4:53:48PM

be at the full rate and that will stop the burst. 4:53:52PM
Q And will that always be more than two cache lines?
A Oh, yes, much more than two cache lines.
Q More than 50 cache lines? 4:54:06PM
A Again, is it 256 cache lines or is it 255 addresses? So assuming 256 cache lines, it will be 256 cache lines, not knowing the fine detail of it if it's 256 addresses, that's a 32-bit address and there are four, so there are eight of those addresses to the cache line. So this would be 32 cache lines. 4:54:33PM
Q So in any case, regardless of what this means, the PCI bus master will still be able to burst across cache boundaries? 4:54:45PM
A No doubt, cache line boundaries.
Q Cache line boundaries?
A Yes, that's why the I2M is 2X the size of what you need. It's called double buffering. I guess it's important to note that this 256 is kind of arbitrary. It could have been a thousand, it could have been 2K. At this time it was, 256 was maybe a recommended off-the-spec or maybe it was a parent data we chose to use but it could be any number. 4:55:05PM

6. The evidence supports Mr. Thome's and Mr. Collins' testimony that they had working silicon, able to practice the MRM command with presnooping and prefetching, in February 1994.

311. As Compaq introduced the CMC/DMC in commercial products in November 1994, it obviously had reduced the presnooping and prefetching invention to practice in prototype silicon much earlier. Mr. Thome, on behalf of Hewlett-Packard/Compaq as a designated corporate representative, testified that Compaq received the prototype CMC/DMC chip, in February 1994. Mr. Thome's notebook supports his contention that Compaq had a working prototype CMC/DMC chip in February 1994.

312. Page HP00039 of Mr. Thome's notebook, EX. 191, dated 2/3/1994, states that "DMC installed . . . no CMC". Mr. Thome, on behalf of Hewlett-Packard/Compaq as a designated corporate representative, testified that Compaq had the DMC chip in silicon as of February 2, 1994:

61

Q. If you could look at the page labeled "HP 39." Has a date 2/3/94 in the upper left-hand corner.
A. Okay.
Q. And there's a figure underneath, and there's some notes to the right of the figure. It says, "CMC CLK. DMC installed. No CMC." And there's something else.
Can you explain the status of -- well, first of all, what is that figure?

62

A. This appears to be some clock signals measured on the oscilloscope that I spoke of earlier.
Q. As of February 3rd, 1994, did you have any silicone for the DMC?
A. We did have DMC silicone, based on what this picture and the notes here say.
Q. Is that consistent with your recollection, as well?
A. It is consistent with my recollection, in terms of roughly where we would have been.
Q. As of February 3rd, 1994, did you have any silicone for the CMC yet?
A. It would appear that it -- it was not arrived yet. And I will say very, very vaguely, it seems like they had a manufacturing error and they had to go back

and re -- redo it, so it wound up taking a little bit longer than originally anticipated.
Q. Were the CMC and DMC on two separate chips?
A. Yes, they are two separate chips.

313. Page HP00041 is dated 2/17 and states "Got started on FUJI Debug." Mr. Thome testified that Compaq had the CMC/DMC chipset in silicon by February 17, 1994, and started on debugging:

62

Q. If you could look at the page that has a Bates label HP 41.
A. Okay.
Q. Is that your handwriting?
A. This is my handwriting.
Q. And there's a note in the upper left-hand

63

corner, "2/17." Does that stand for February 17th, 2004?
A. It would not be 2004. It would have been --
Q. I'm sorry. 19 -- does that note stand for February 17th, 1994?
A. I believe this would be February 17th, 1994, because, sequentially, it would have been that year.
Q. And there's a time stamp or a time note right next to it, 11:45. Do you see that?
A. Yes.
Q. And then you write, "Got started on Fugi debug." Do you see that?
A. Yes.
Q. What does that mean?
A. That means that we received all the silicone and manufactured the Fugi board and brought it in the lab and plugged it in and, of course, it didn't work, so we were starting to figure out why it wasn't working correctly.
Q. As of February 17th, 1994, did you have silicone for the CMC and DMC?
A. I -- I believe so, yes.
Yeah, I have notes here saying that we were -- where is it -- the CMC was installed.
THE REPORTER: I'm sorry?

64

A. I have a note here implying -- implying that the CMC would have been installed.
Q. (BY MR. TETER) Can you point out that note?

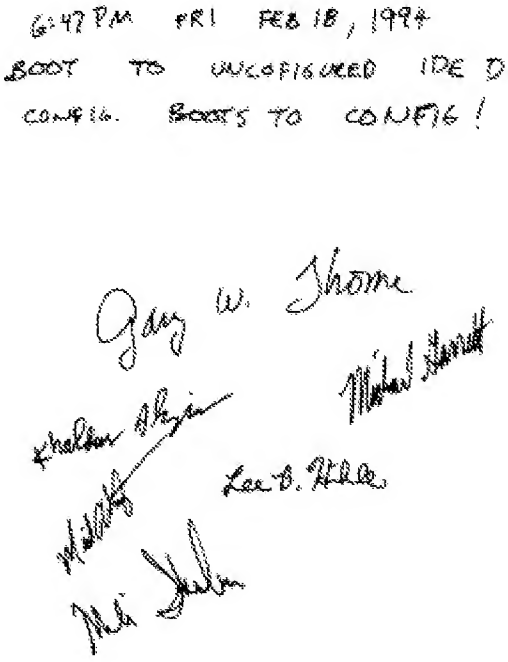
A. At the very bottom, it says -- where it says, "Problem: This debug board generates the wrong PC address out of the CMC on the first ROM fetch."

So the only way that the CMC could generate the wrong address is if the CMC was installed on the board.

Q. So as of February 17, 1994, Compaq, then, had both the CMC and DMC for the CMC/DMC chipset that you described earlier?

A. It would appear that way, yes.

314. Page HP00044 is dated 6:47pm, 2/18/1994, and states "Boot to unconfigured IDE drive with config. Boots to config!" with six different signatures.



[EX. 191 at HP00044].

315. Mr. Thome testified that Compaq had successfully booted an operating system on the CMC/DMC chipset on February 18, 1994:

64

Q. If you could turn to the Page HP 44?

A. Okay.

Q. And there's a time in the upper left-hand corner, "6:47 p.m., Friday, February 18, 1994." Do you see that?

A. I do.

Q. And then it says, "Boot to unconfigured IDE drive with config boots to config." Do you see that?

22 A. Yes.
23 Q. And a bunch of signatures underneath. Do you
24 see that?
25 A. Yes.

65

1 Q. What is that note, "boot to unconfig"? What
2 does that mean?
3 A. So it was -- so once we got a first prototype
4 in, it was a bit of a celebration when we could actually
5 get a boot to an operating system. So that meant that
6 it would work sufficiently well enough that it could
7 execute code and get to the point that it could load --
8 like, in this case, it probably would have been
9 Microsoft DOS.
10 And so when we got to that point, we would
11 always -- as soon as it happened, we would write down on
12 our lab notebook that it happened and say exactly what
13 happened, the exact time and then all of us who were in
14 the lab working on it at the time would sign it, just as
15 kind of a -- I guess, a remembrance, more -- more of
16 anything. It was just a happy moment.
17 Q. So as of February 18th, 1994, at 6:47 p.m.,
18 then Compaq had silicone for the CMC and DMC, and it had
19 successfully booted up an operating system?
20 A. That's correct.

316. Mr. Thome testified that over the next several months, Compaq debugged the silicon and worked to make a salable product.

65

21 Q. What did Compaq do after it succeeded in
22 booting up the CMC/DMC with an operating system? What
23 happened over the next couple of months?
24 A. The next several months, we started running a
25 number of different tests on it, and attempted to

66

1 identify any bugs that might have existed in the
2 silicone, corrected those bugs and as well as other
3 bugs, such as some of the software code or firmware
4 code, what we call signal integrity, which is measuring
5 the signals, make sure they're running at the right
6 speed that they're supposed to run at on the board.
7 So there was a number of different tests
8 and procedures that we run over the time to -- to get it
9 fully operational to the point that we could then make
10 it a salable product.

317. Page HP00049 is dated 2/20 and states that "FUJI now boots with L1 and L2 enabled." Mr. Thome testified that Compaq had successfully booted with both caches enabled by February 18, 1994:

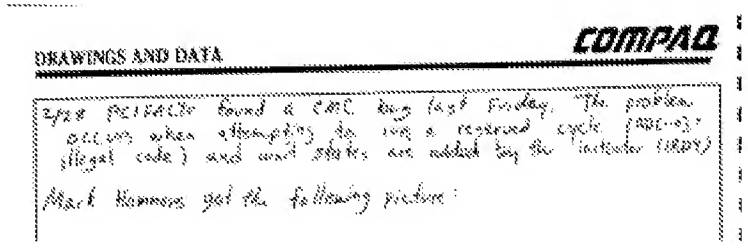
66

Q. If you could turn to the page labeled "HP 49."
A. Okay.
Q. There's a note at the bottom. It states, "Fugi now boots with L1 and L2 enabled," three exclamation points, remarks after that?
A. Yes.
Q. And a date and time, "3:15 p.m. 2/20." Do you see that?
A. Yes.
Q. What does that mean?
A. So this means that now -- so when we got it booting, originally, it was booting without the -- the caches enabled. And the caches provided some additional complexity in the system because they -- you know, so having them turned off made it easier to get it running.

67

What this means is, that we now got it to boot an operating system with the caches turned on. Both -- the L1 stands for the cache that's inside the microprocessor, and the L2 is the cache that's controlled by the CMC chip.
So that means that we got it to boot up to an operating system with those caches turned on, and that was a good thing.

318. Page HP00062 states that on February 28, Compaq was running PCIFACTs and looking for bugs.


DRAWINGS AND DATA COMPAQ

[EX. 192 at HP00072].

319. Mr. Thome testified that PCIFACTs was a tool that Compaq used to test a product with every PCI command.

67

Q. If you could turn to the page labeled "HP 62."
A. Okay.
Q. There is a date in the upper left-hand corner of 2/28?
A. Yes.
Q. See that?

68

A. Uh-huh.
Q. It says, "PCIFACTs found a CMC bug last Friday. The problem occurs when attempting to run a reserve cycle..." and then it goes on.
First of all, is that your handwriting?
A. That is my handwriting.
Q. What does that note mean, the 2/28 PCIFACTs note that we just read?
A. So 2/28 would have been the date, so it was February 28th. That --
Q. What year?
A. Of 1994, that I made the entry into this.
PCIFACTs is a -- is a tool that we used to test out our computers and verify their functionality. And so that -- so my note here was that the PCIFACTs found a combination of cycles that resulted in erroneous operation of the CMC.
Q. So in February 1994, you were debugging the CMC and DMC and actual silicone; is that right?
A. That's correct, yes.

320. MRM was, of course, a PCI command as of February 1994, and had been so for nearly a year. Compaq, as a member of the PCI SIG, obviously was aware of that, and in August 1993, Compaq was already discussing the high performance that it planned for the MRM command (as compared to the MRL command), which Compaq would achieve by presnooping and prefetching. [EX. 188].

321. Mr. Thome testified that in 1993, Compaq planned to achieve that performance by prefetching and presnooping.

46

Q. In August 1993, what did Compaq anticipate the performance of the CMC would be for the Memory Read Line

for the 33 megahertz processor?
A. I believe it would be 67 megabytes per second.
Q. And what did Compaq anticipate at that time would be the performance of the CMC for the Memory Read Multiple command?
A. 129 megabytes per second.
Q. Why at that time did Compaq anticipate that there would be such a big improvement in the performance for Memory Read Multiple versus the performance for Memory Read Line?
A. Because in a Memory Read Multiple command, we could pipeline the -- the memory reads one after another, to continue to sustain the -- the peak

47

bandwidth.
THE REPORTER: The -- I'm sorry, the what?
THE WITNESS: Continue the sustained bandwidth, sustained peak bandwidth.
Q. (BY MR. TETER) And is that the process of presnooping and prefetching that you were discussing before?
A. It was a process of snooping and a -- and a -- and reading ahead the -- the main memory, yes.
Q. So as of August 1993, Compaq anticipated performing that process and achieving 129 megabytes per second per Memory Read Multiple command; is that right?
A. That is correct.

322. The PCIFACTS tests support Mr. Thome's testimony that the CMC/DMC chipset was able to run MRM cycles successfully in the February 1994 time frame.

109

Q. I didn't see anything in here like that to reflect when you successfully executed the MRM command on the chipset. Is -- is there an entry of that sort in here? And I'm referring to Exhibits 191 and 192.
A. Well, so I don't think -- so based on what we've looked at and me skimming through the notes, I don't see anything specifically like that. But if you recall, we talked about something earlier about PCIFACTs running. I did have a note in there about PCIFACTs running?
Q. Uh-huh.
A. And the way PCIFACTs was designed, it was designed to actually iterate through every single possible PCI command that could be run.
And so -- so, you know, to the extent that MRM was, you know, a PCI command, then we would have been implementing it with the PCIFACTs tool and we would have tested it.

110

And so the notebook, I was primarily making notes for when we found problems. The one exception, as I mentioned earlier is, yeah, we get excited when we boot so we, you know, make a note of that --

Q. Uh-huh.

A. -- kind of a thing.

So we -- you know, there's -- you know, there's PCI, there's PCI read, PCI Read Multiple, PCI write, you know. And there's a whole bunch of different commands and we wouldn't make a note every single time we -- we got one working or not working.

110

Q. If I understand you correctly, then, what you're telling me is that your recollection is that you had some software that was running those tests but there's no written record to confirm that -- that the tests came back actually showing the MRM command operational?

MR. TETER: Object to form.

A. So -- so I think what I'd say is, my recollection is that we ran the PCIFACTs and, you know, we would have some notes in here showing, you know, when

111

we were running it and when we found a -- a particular bug.

The way we would do it is, we'd sit down with the PCIFACTs tool, and first we would run through every different kind of command. Then we would run through -- then we'd let it run -- iterate through all the different variances of that command.

So it would like run, you know, a cycle at offset one byte in the cache and offset two bytes in the cache, offset three bytes. And so we'd kind of iterate through all those combinations to make sure that they all worked properly. And so there were thousands upon thousands of different possible combinations. And we would use that software and verify the operation of the -- of the design that way.

And then the meat grinder software that I -- I spoke of earlier also used the same hardware cards to generate all the different kinds of PCI bus cycles, all the different PCI traffic. And that tool would be one where we would then let them all run simultaneously and you'd generate a huge amount of fairly random traffic just to try to find some of the kind of the -- the corner case bugs, primarily.

323. Mr. Thome's debugging notebooks indicate that he found, and solved, bugs in the CMC/DMC chipset.

324. Mr. Thome testified that the CMC/DMC chipset worked as anticipated for the vast majority of cases, and that the bugs appeared only in the very rare combinations of events.

77

Q. Now, earlier you testified about how the CMC/DMC would do prefetching and presnooping in a Memory Read Multiple command?
A. Yes.
Q. You described it as like pipelining, right?
A. Uh-huh. Yes.
Q. For the basic scenario where a PCI master is just reading from secondary memory and there aren't any other requests going on by the processor or anyone else, did the CMC/DMC silicone that you received in February 1994 work for that basic process?
MR. BRODY: Object to the form.
Q. (BY MR. TETER) Do you understand the question?
A. I do understand the question.
So -- so I guess what I would say is that the -- the CMC that we received worked correctly for nearly every combination. Bear in mind, there were probably millions of different possible combi -- combinations, and there were -- the ones that didn't work were probably in the -- in the tens or maybe hundreds, at most, of combinations.
So nearly everything did work correctly as -- as we expected or anticipated it to.

78

Q. This bug that you identified, is this what you would sometimes describe as a "corner case"?
A. Yeah. So we would say it was an unusual, you know, sequence, and it took an awful lot of effort to even -- to get it to occur. And so it was one that was -- you know, you could -- you could describe as a corner case, although it was certainly important to have it fixed before we took it to production because, otherwise, you would -- you know, even -- even though it only happened infrequently, infrequently having the wrong data is not a good thing for customer applications.
Q. So for most case, the CMC/DMC was able to perform the prefetching and presnooping, that you described before, at least as it -- as you had the silicone in early 1994; is that right?
A. Yeah.
MR. BRODY: Object to the form.
A. Okay. So the fact that it took us three months

20 before we saw the bug, it was clearly operating
21 correctly for most cases..

325. The bugs that Compaq identified and solved in the CMC/DMC chipset were for what I have described above as "corner cases" in a design. The entries in the Compaq notebook do not indicate any problem with the PCI MRM command, which tends to indicate that the command was working as anticipated. If the PCI MRM command had *not* been functioning, one would have expected to see many entries about it. Mr. Thome, on behalf of Hewlett-Packard/Compaq as a designated corporate representative, testified:

133

17 Q. You mention that there was a -- a program that
18 you had called PCIFACTs. Do you remember mentioning
19 that?
20 A. Yes.
21 Q. If you look at Page HP 62 of Exhibit 192?
22 A. Okay. I'm here.
23 Q. And you have the note, "2/28, PCIFACTs found a
24 CMC bug last Friday." Do you see that?
25 A. Yes.

134

1 Q. Does that indicate that you were running
2 PCIFACTs on the silicone in February 1994?
3 A. Yes, it does.
4 Q. Now, if you had had or found a bug -- strike
5 that.
6 Did -- did PCIFACTs run tests on all PCI
7 commands?
8 A. Yes, it did.
9 Q. In 1994, did PCIFACTs then run a test on MRM?
10 MR. BRODY: Object to the form.
11 A. I believe that we'd run tests for MRM. And I'm
12 just looking at the timing diagram right here -- and I
13 have to refresh my memory -- but it may actually even be
14 a MRM command that's on the timing diagram right here.
15 Q. (BY MR. TETER) If you had found a bug, in
16 February 1994, on the silicone for the MRM command,
17 would you have recorded it in your engineering notebook?
18 A. Likely. Probably would have, but I attempted
19 to record the bugs as we found them, in the notebook.
20 Q. And the notebook doesn't indicate any
21 particular bug for MRM, does it, not that you saw in
22 your review last night?
23 A. I don't recall seeing anything particular call
24 out a MRM.
25 Q. Is it consistent with your recollection that

135

the MRM command in the silicone that you received in 1994 February operated as expected?
MR. BRODY: Object to the form.
Q. (BY MR. TETER) Do you understand the question?
A. Yes. So my recollection is that it generally operated correctly with the MRM command.
Q. And when we say "operated correctly," are you referring to doing the prefetching and presnooping that you described earlier on direct?
MR. BRODY: Object to the form.
A. So my recollection is that it would have done the presnooping and prefetching.
Q. (BY MR. TETER) As of February 1994?
A. As of February of 1994, yes.

326. Thus, the written evidence, including the Compaq lab notebooks and whitepaper, support the testimony of Mr. Thome, on behalf of Hewlett-Packard/Compaq as a designated corporate representative, that Compaq had reduced the presnooping and prefetching invention to practice in February 1994.

327. In 1994, after debugging the silicon for the prototype, Compaq had the second version of the silicon, CMC-2, manufactured.

68

Q. If you could look at the bottom line of Page HP 62?
A. Yes.
Q. It reads, "No CMC-1 workaround; will be fixed in CMC-2." Do you see that?

69

A. Yes.
Q. What was the difference between CMC-1 and CMC-2?
A. Well, CMC-1 was the -- the first version of the chip that we used to bring in the lab and start debugging. And as we found bugs in it or errata, we would make those corrections and then we'd manufacture the second version of -- of the CMC chip, which was called CMC-2. And that second version is the one that we ultimately took to production, if I recall.
Q. Now, when you talk about these two different versions, CMC-1 and CMC-2, does that correspond to what you were describing before as -1 or A0 samples in -2 or A1 samples, in Exhibit 188?

A. Yes. So we always use the terminology -1 and -2 for our versions.

Q. When did Compaq receive CMC-2?
A. It would have been sometime in 1994, and I don't recall the exact date.

328. Supported by his lab notebooks, Mr. Thome testified that Compaq received the CMC-2 silicon sometime in August 1994.

107

Q. Okay. What happened between May 24th and August 30th, do you know, on the CMC/DMC?
A. Well, so I -- I don't recall specifically, but I will say that probably what happened here is, this is probably we -- this was probably the last bug that we

108

found in the CMC-1. And sometime a little bit after that, we released CMC-2. And as I mentioned, it seemed -- you know, it takes several weeks, you know, a couple months to -- once you release the silicone before you actually get it back.

So if I were to look at this, I'd say we probably released CMC-2 probably sometime in -- in June, and then started receiving the silicone back in August, started building up more boards.

And then the -- the notes on August 30th is -- is reflective of what we called our wide area test, where we go from kind of our engineering lab, the engineering development lab and start moving it into a number of other test labs where they test a lot of -- a much broader set of -- of third-party adapter cards, third-party software, and so forth like that.

And this -- that's the kind of work I was doing at the end of August.
Q. Okay.
A. That's -- that's a little bit of speculation on my part. But it looks to me, based on the information and the time -- timeline that would probably have been roughly accurate.

329. Mr. Thome testified that it took Compaq approximately 2 months after determining that a chipset was ready for production to include it in a commercial product.

44

From the time that a chipset is ready to be produced, to the time it actually ends up in products, what's required -- or what was required at Compaq?

A. We would get the -- you know, the second version of the prototype in or whatever the production version of the -- the silicone in, and we would test it for -- for a -- usually some amount of time and it would -- depending on how substantial the changes were, could range from a couple weeks to a few months.

And once we felt confident that it was going to work, we would start the -- we would place an order to our production fabrication facility that would make the production versions of the chips, and that would usually take several weeks to, say, maybe about -- maybe about six weeks before they get the first version for those completed. And then it would take us another, roughly, two weeks to put them on motherboards, put them into computers, and then start shipping the product to customers.

330. Mr. Thome's timeline is completely consistent with the launch of the CMC/DMC in November 1994 indicated by the Compaq/HP documents.

7. Mr. Thome and Mr. Collins conceived of the presnooping and prefetching invention in the 1992-93 time frame, before OPTi.

331. Mr. Collins and Mr. Thome testified that they conceived of the presnooping and prefetching invention early in the design of the CMC/DMC chipset.

332. Mr. Thome, on behalf of Hewlett-Packard/Compaq as a designated corporate representative, testified:

8

Q. When did you start working on the CMC/DMC?
A. It was in 1992.

53

Q. (BY MR. TETER) By what date had Compaq decided to have an I2M queue buffer that would store two cache lines of data?
A. I don't recall the exact date, but it would have been either sometime in the second half of 1992, or the -- probably the -- or possibility the first half of 1993.
Q. By what date had Compaq decided to do the prefetching and presnooping, that you described earlier, in the context of a Memory Read Multiple command?
MR. BRODY: Object to the form.
Q. (BY MR. TETER) Do you understand the question?
A. I do understand what the question is.

22 So -- you know, so I have to base this off
23 of my recollection, and I believe it would have probably
24 been in the second half of 1992.

333. Mr. Collins testified:

111

24 Q Do you recall whether the triflex PCI
25 architecture implemented presnoop in 1992? 5:15:02PM

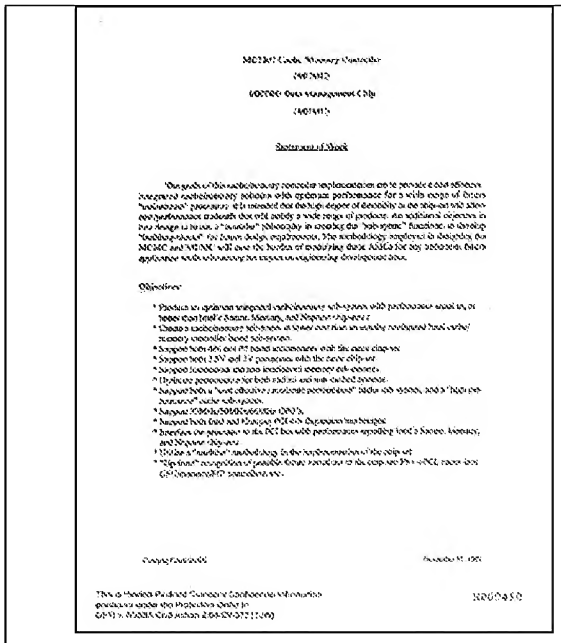
112

1 A What's your definition of presnoop? 5:15:10PM
2 Q Do you recall whether the triflex PCI
3 architecture contemplated snooping the next line
4 while the current line was being received?
5 A Yes, being sent out. 5:15:22PM
6 Q Let me ask it again, complete the
7 question. Do you recall whether the triflex PCI
8 architecture snooped the next line while the current
9 line was being received by the PCI bus master?
10 A Yes. 5:15:34PM
11 Q And it did that as of November 11th, 1992?
12 A Again, I can't vouch for the date.
13 Q If this date is accurate?
14 A If the date is accurate, that was clearly
15 in the design of this. 5:15:46PM

187

5 Q You're confident as of September 24th, 9:05:39PM
6 1993, Compaq had implemented a system that allowed
7 PCI bus masters to burst across cache line
8 boundaries?
9 MR. BRODY: Object to form.
10 THE WITNESS: Again, not recalling the 9:05:57PM
11 exact detail, I believe so. I mean this is on the
12 order of maybe almost a year after this project.
13 And like I said earlier, 183 is a follow-on to the
14 DMC. This is the Biner chips. So this is the
15 second generation data controllers for ECC. 9:06:18PM

334. The earliest document describing the project, the "METRO Cache/Memory Controller . . . Statement of Work" indicates that Compaq was planning to include read prefetch buffers in November 1992.

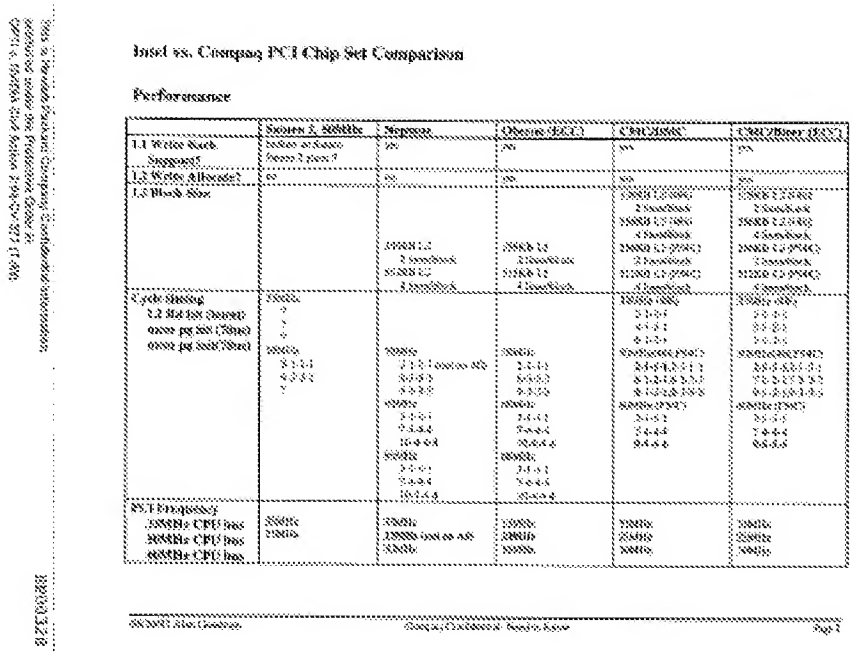


Proposed Architectures:

* 3-chip solution for P54 / 2-chip solution for 486
* Sub-micron CMOS technology
* Integrated tag RAM for speed/cost optimization
* 3.3V or 5V CPU interface / 5V PCI interface
* Serial architecture with Writeback 1.1 / Writeback 1.3 protocol
* 33/50/66 Mhz CPU support
* 0-wait-state read hits at 33/50 Mhz
* Programmable wait states for cache and DRAM accesses
* Upgradeable cache: 128K, direct-mapped -> 256K, 2-way for 486 architecture;
  or 256K, direct-mapped -> 512K, 2-way for P5 architecture
* Support for both standard and burst cache data RAMs
* Direct-mapped cache modes are 2 lines per block;
  2-way cache modes are 4 lines per block.
* Support for up to 144 MB of local DRAM with flexibility in configurations
* Unbuffered RAS and CAS lines. MA and WE buffers require buffering.
* 8-deep cpu-to-memory and 4-deep cpu-to-pci posting buffers.
* PCI read prefetch buffer.
* PCI bus speeds of 25/33 Mhz, synchronous to the CPU clock.
* Fully concurrent PCI implementation isolates the CPU.

[EX. 185 at HP00412].

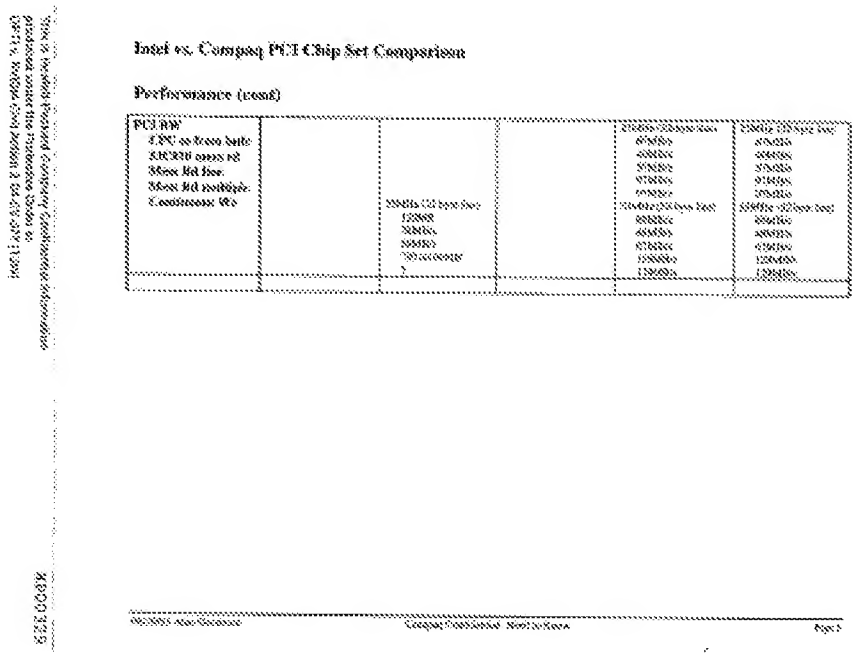
335. The conception of the Triflex/PCI invention occurred earlier than August 1993, as the internal documents at Compaq show that the Compaq invention had already been conceived and would perform a PCI memory read multiple (MRM) at 129 Mbytes/sec, which asymptotically approaches the maximum throughput of the PCI bus.



Intel vs. Compaq PCI Chip Set Comparison

Performance

[EX. 188].



[EX. 188].

336. To achieve such bandwidth, which requires a data unit to be sent every PCI clock cycle, the Compaq inventors had conceived of the pipelining (prefetching and presnooping) that later appeared in the Compaq whitepaper. Mr. Collins testified that the presnooping and prefetching allowed the Triflex/PCI chipset to send a data unit every clock cycle:

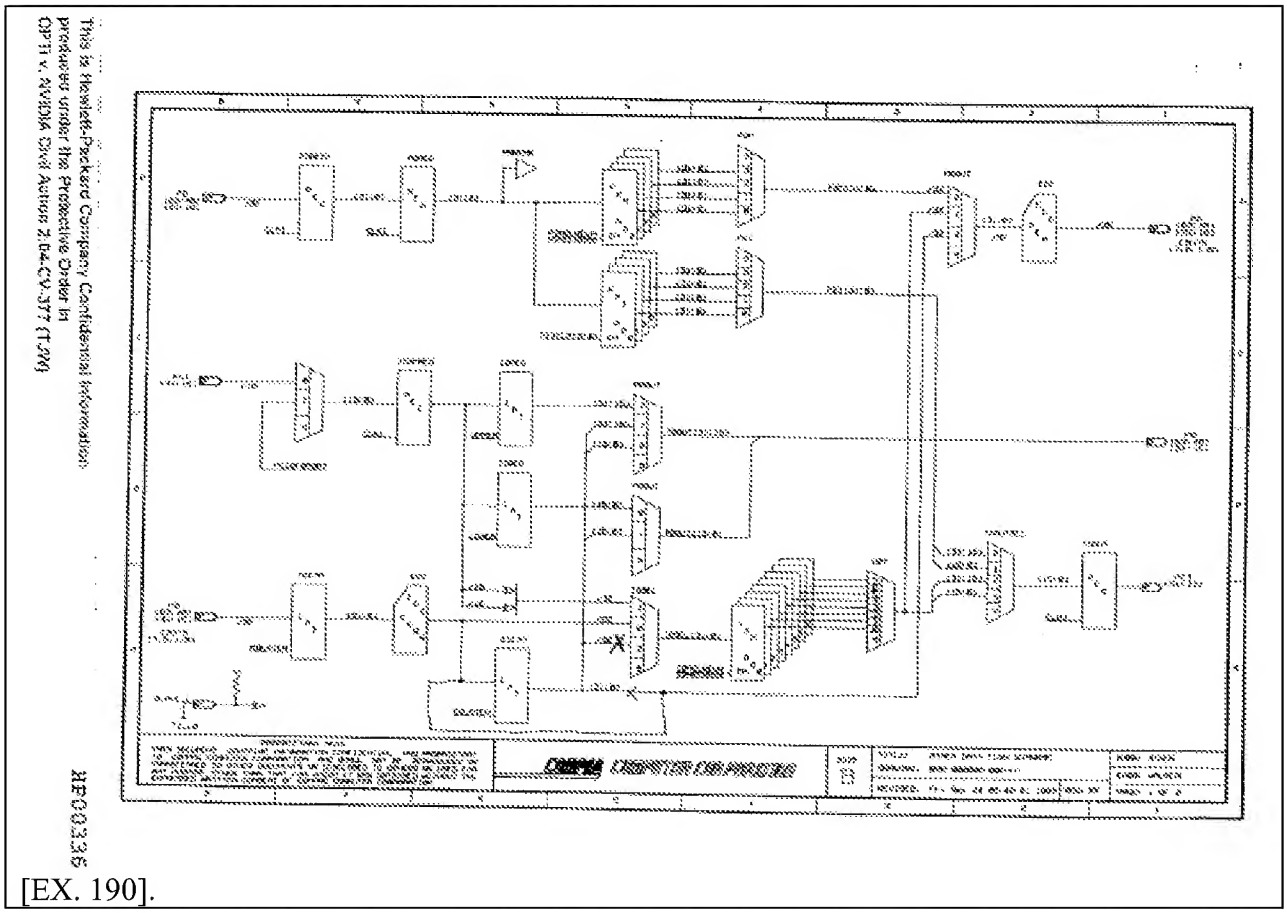
86

Q In normal operation would the triflex PCI architecture presnoop in response to a memory read multiple command? 4:22:44PM
A It will try.
Q And what effect would this presnooping and prefetching have on the data flow from the perspective of the PCI bus master?
A It will be bursting data. It won't be 4:22:58PM interrupted. So, for example, on the PCI bus, every PCI clock there can be a data transfer transaction. If both the master and the slave are enabled with the readies, you can move data with every clock. It would be the goal of the triflex chip to move data 4:23:23PM every clock.
Q So the results with respect to the presnooping and prefetching is the that PCI bus master would receive a constant rate of data?
A Yes. Well, every clock there will be a 4:23:37PM

21 data transfer, yes.
22 Q Will there with a data transfer across
23 cache line boundaries?
24 A Yes, of course because this is two cache
25 lines, so that's the whole reason the queue is 8. 4:23:53PM

337. Mr. Thome testified that Compaq already had conceived of the presnooping and prefetching method by August 1993, the date of EX. 188.

338. In addition, a Compaq drawing of the Biner chipset (which used the same architecture as CMC/DMC) dated September 24, 1993, shows the Compaq I2M queue, and illustrates that it can hold two cache lines of data:



[EX. 190].

339. As noted above, having buffers that are able to hold two cache lines of data strongly supports Mr. Thome's and Mr. Collins' testimony that they conceived of the presnoop and prefetching idea from the very early stages of the process. In fact, Figure 5 of the patent is virtually identical to the September 1993 Compaq drawing.

340. Furthermore, in order to have silicon in February, the Compaq engineers would have completed the design, and taped it out, earlier.

8. The documents that OPTi cites do not contradict Compaq's conception.

341. During the depositions of the Compaq inventors, OPTi's counsel argued that typewritten documents dated in late 1992 / early 1993 contradicted the testimony of Mr. Collins and Mr. Thome. According to the argument of OPTi counsel, the typewritten documents showed that Compaq planned to disconnect at cache line boundaries and that Compaq did not plan to have a streaming feature.

342. Neither Mr. Collins nor Mr. Thome knew who wrote the typewritten documents or what exactly they represented. Mr. Collins speculated that Michael Moriarty might have written the document, but nevertheless remained confident that the Triflex/PCI architecture permitted PCI bus masters to burst across cache lines by 1993:

188

Q Are you confident that -- do you know who authored Plaintiff's Exhibit 100?
A This could be Michael Moriarty.
Q Why do you believe that Michael Moriarty might have authored Plaintiff's Exhibit 100? 9:08:42PM
A Because these are very specific to the memory, excuse me, to the PCI reads and writes.
Q This is dated in 1992, correct?
A Yes. 9:09:05PM

189

Q Are you confident that as of 1993 the triflex PCI architecture permitted PCI bus masters to burst across cache line boundaries? 9:09:06PM
A Yes.

343. Responding to questions from OPTi's counsel, Mr. Thome testified that he recalled always wanting to do read ahead (prefetching and presnooping) from the start, and emphasized that it was included in the February 1994 silicon:

103

Q. And the reason I wanted to point you to that passage was to ask you whether that indicates that the plan, in December of '92 at least, was to do bursts only of one cache line at a time?
A. So in my recollection, I don't recall ever making a decision that we would do that. The -- the passage here would seem to indicate that that was what our thinking was at the time or --
Q. Okay.
A. -- whoever wrote the document, that was their thinking at the time. I -- I was not the -- the author of the document.
Q. Okay.
A.[sic, Q] But your recollection is that, eventually, you changed that, and you decided to burst more than one cache line at a time?
MR. TETER: Object to form.
A. So -- so my recollection is, as long as I can remember discussing it, we had decided to -- to burst it. But...
Q. (BY MR. BRODY) Burst more than one cache line?
A. Burst more than one cache line on read -- on

104

Memory Read Multiple commands.
But as I said, you know, I was not the only person on the -- on the design, so it's certainly possible that either we were all thinking this at -- at the time or some people were thinking at the time and not all of us.
Q. Do you know when the group came to consensus that you were going to burst more than one cache line? Can you give me a date?
MR. TETER: Object to form.
A. Well, so it would have been before we re -- received the CMC-1, because it was part of the feature set from --
Q. (BY MR. BRODY) Okay.
A. -- from CMC-1.
So it would have been sometime relatively early in the design cycle of the CMC-1.
Q. Okay. You got the CMC-1, and I think we were looking at February or March of '94?
A. Yes.

344. Mr. Collins also disagreed with the interpretation of the documents by OPTi counsel, despite a lengthy examination. Mr. Collins testified that he believed that the documents were describing bursting in another context and that the document did not affect his

belief that, as of 1993, the Triflex/PCI architecture permitted PCI bus masters to burst across cache line boundaries.

170

Q Isn't that possible?
A Anything is possible, but I cannot imagine 8:28:22PM
that we would be patenting a concept that's not in the chip.
Q The fact that you ultimately got sometime in 1994 to multiple lines of buffers, multiple line buffers, I should say, and that that's, may well 8:28:39PM

171

have gone into the chip and may have gone into your 8:28:43PM
patent that it's not inconsistent with the fact that earlier on you may have had a different architecture in mind?
MR. CUNNINGHAM: Object to form. 8:28:54PM
THE WITNESS: Again, I'm disagreeing with you because I think we are reading the way this is as bursting as not being the meaning that we were doing. I think you're trying to -- it's the PCI burst and I think it is really the fact that the 8:29:06PM
lead-off address of a PCI read would be always reading one line or not.
So, for example, this could be a memory burst that a PCI is one 32-bit address, but the memory system out back is eight times that. So we 8:29:27PM
would be doing the bursting but not of more than one line which would be eight times the data of what the PCI's asking for. I think that's where I disagree. It's not the PCI burst.

186

Q If I could direct you to Exhibit 168 which is the schematic that is Bates labeled HP336. Are you confident that the triflex PCI architecture permitted PCI bus masters to burst across cache line 9:05:03PM
boundaries?
A Yes.
Q And are you confident that Exhibit 186 illustrates the implementation of the architecture that permitted PCI bus masters to burst across cache 9:05:21PM

187

line boundaries? 9:05:25PM
A From a data path perspective, the data path is here for two cache lines to be double buffering.

Q You're confident as of September 24th, 9:05:39PM
1993, Compaq had implemented a system that allowed
PCI bus masters to burst across cache line
boundaries?
MR. BRODY: Object to form.
THE WITNESS: Again, not recalling the 9:05:57PM
exact detail, I believe so. I mean this is on the
order of maybe almost a year after this project.
And like I said earlier, 183 is a follow-on to the
DMC. This is the Biner chips. So this is the
second generation data controllers for ECC. 9:06:18PM
BY MR. CUNNINGHAM:
Q You have no reason to believe that the
date on Exhibit 186 is inaccurate, do you?
A No, I don't.

345. Mr. Collins only caveat was that the Triflex/PCI architecture, in 1993, might have had a restriction on bursting across cache lines for a write-through (not write back) cache—which is a different context that does not involve snooping and is not relevant:

189

Q And as of 1993 the triflex PCI
architecture would presnoop and prefetch in response
to a PCI memory read multiple command?
MR. BRODY: Object to form. 9:09:28PM
THE WITNESS: To answer, with limitation.
I think what was pointed out earlier there seems to
be some restriction on write-through address tags.
Now I've lost my hundred percent confidence.

346. Even if the Triflex/PCI architecture did have a restriction on bursting in the write-through context in 1993, it would still have burst across cache lines when operating with a Pentium microprocessor because the Pentium microprocessor utilized a write-back cache. Thus, Mr. Collins remained confident even at the very end of his deposition that the Triflex/PCI architecture permitted PCI bus masters to burst across cache line boundaries:

189

Q Are you confident that as of 1993 the 9:09:06PM
triflex PCI architecture permitted PCI bus masters
to burst across cache line boundaries?

347. Mr. Thome testified that the product as shipped did the read-ahead (presnooping and prefetching) for all PCI MRMs.

Q. Do you recall that in December of 1992, you were going to prefetch for some PCI to memory reads but not for others?
A. So -- so, yes, we wrote this down, presuming this is what we were doing. I don't recall what -- you know, what the distinctions were, in terms of when we would do the prefetching and when we would not.
Q. Okay. Do you know that -- if that's the way you ultimately did it?
A. The way we ultimately did it -- so the configuration registers were used to be able to -- primarily to do things like either set a -- set a setting for something or to be able to turn features on or off.
Q. Uh-huh.
A. So -- and as I can vaguely recall, I believe that we probably had a configuration registered to be able to turn off the feature of being able to -- to read multiple cycles or, you know, read the -- you know, prefetch and read multiple cycles.
And so I'm -- I'm kind of saying this from

96

really a vague recollection because I don't recall the specifics of -- of how we implemented exactly. But we probably had something to turn that off because that would have been a feature we would have been concerned about possibly having bugs in. And so we said, well, we can turn it off so that we can get around it, if we need to, to continue debugging other functions of the -- of the chipset.
Though, when we shipped it, we shipped it with PCI Read Multiples always reading ahead.
Q. Okay.
A. And the PCI standard read cycle, we may have had a configuration register option to decide whether or not to prefetch or not prefetch.
As I -- as I recall, I think our concern was it was unclear for -- so it was unclear because the PCI specification was new at the time. And so it was unclear if -- if PCI bus master devices would issue PCI Read Multiple commands, when what they really meant to -- or issue PCI read commands when what they really meant to do is issue PCI Read Multiple -- Multiple commands.
This is coming back to me slowly here. So I think what we did is, we had a register to say if we found a PCI device that was running PCI read commands

97

when they really -- what they really wanted was to issue

PCI Read Multiple commands, we would treat that PCI read command as if it was a PCI Read Multiple command and run the prefetch cycles for that device on -- on its behalf. In effect, treat it like it was a Read Multiple command, even though it wasn't.

And this is still a recollection, so we'd have to look at it much closer.

97

Q. Sure. Do you know when you got to the point where you decided to do all the read multiple commands with prefetches?

MR. TETER: Object to form.

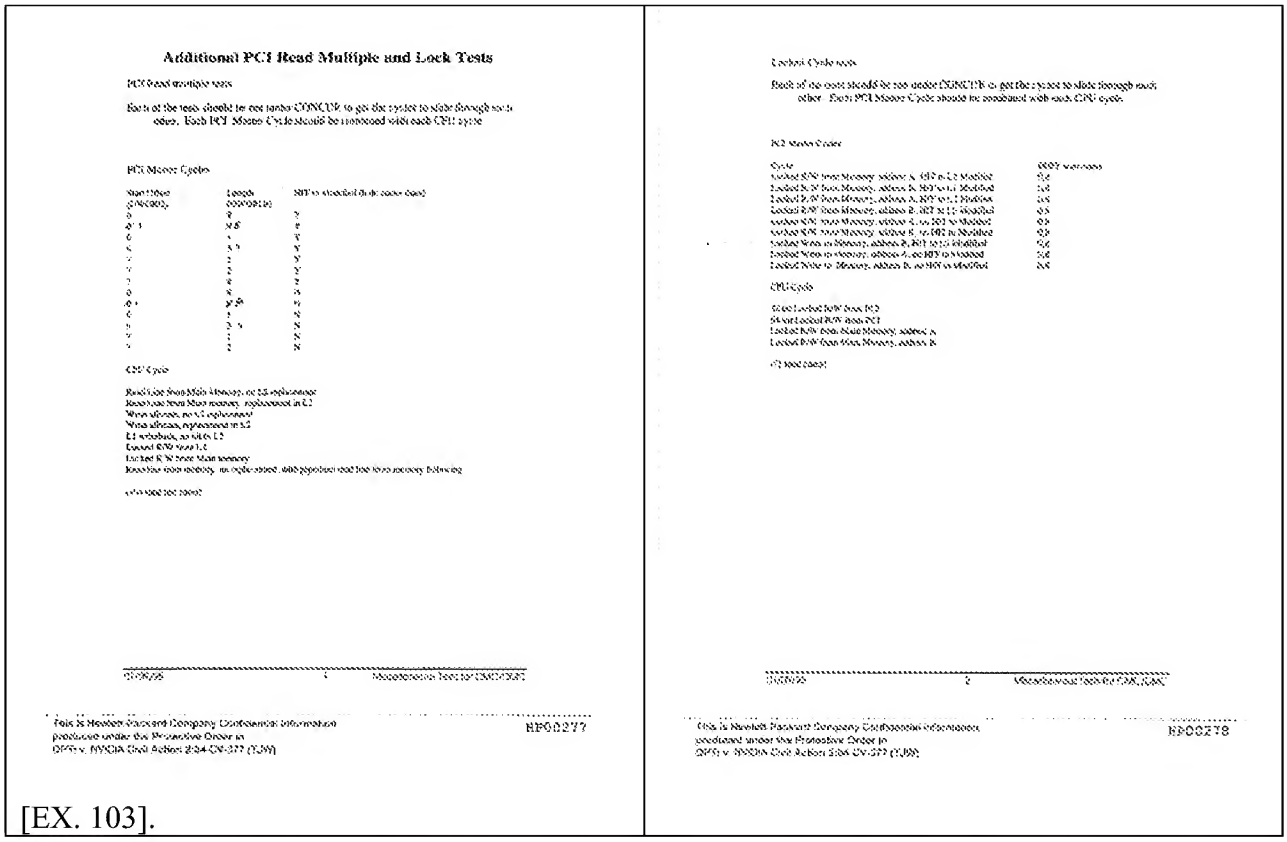
A. It would have been fairly early in the -- the development cycle that it was pretty clear from the PCI specification that the read multiple command was intended to provide a hint to us that the PCI device was intending to continue reading.

And so it made obvious sense to us that we would, in the process of developing, want to go ahead and -- and read ahead.

348. Even if the typewritten documents were read as OPTi counsel interprets them, they are not relevant under the jury instructions. Mr. Thome and Mr. Collins, the inventors, planned to burst across cache line boundaries from the outset. Mr. Thome testified that at most someone may have for a time proposed a different approach, but the inventors, Mr. Thome and Mr. Collins, conceived of bursting across cache line boundaries.

9. The documents that OPTi cites do not contradict Compaq's reduction to practice.

349. During the depositions of the Compaq inventors, OPTi's counsel argued that a Compaq document, entitled "Additional PCI Read Multiple and Lock Tests" shows that Compaq had not reduced the MRM method to practice even as late as January 1995, the date printed on the document. The document cited by OPTi appears below.



Additional PCI Read Multiple and Lock Tests

HP00277

HP00278

[EX. 103].

350. OPTi counsel showed the document to HP/Compaq's designated representative, who testified that the date at the bottom may well be a printed date, not the date that the document was prepared, and testified that the document did not contradict his testimony that the MRM prefetch/presnoop invention had been reduced to practice as early as February 1994. Mr. Thome testified:

123

13 Q. Okay. And the document that Exhibit 1 -- the
14 date, I'm sorry, that Exhibit 103 bears is January 9 of
15 1995; is that right?
16 A. Well, so the -- so the document has the date
17 1/9/95 printed on it.
18 Q. Okay. Do you have any reason to think that the
19 document was generated on any other date? Or let me ask
20 you this: Do you know what the date of the document
21 is --
22 A. Well, so --
23 Q. -- beyond what appears on its face?
24 A. So I would presume that this -- this document
25 was generated on 1/9/95, but there were probably earlier

versions of the document, also.
Q. Okay.
A. Since the document -- documented -- we would have documented the -- the tests that we were doing probably over a series of time, but I don't recall the particulars of it.
Q. And then Exhibit 103 would appear to indicate that, as of January 1995, these -- these PCI Memory Read Multiple tests were going to be run at some future date under the CONCUR program, right?
MR. TETER: Object to form.
A. Or they were already previously run.
Q. (BY MR. TETER) Well, if you look at the -- the first line of text, it says, "PCI" -- it says, "PCI Read Multiple tests."
And then it says, "Each of the tests should believe run under CONCUR." See that?
A. Yeah.
Q. Okay. So that's saying, in the future, they're going to be run under CONCUR?
A. Well, as I -- I guess that's what I'm -- I'm not saying this very well. But whenever this document was written and whoever wrote it -- and I don't even remember who wrote it -- and it may have even been me, I don't remember -- when they wrote the document, they

would have written it in the future tense because they were writing, saying these are the things that need to be done. But as the document gets iterated and printed out, the text probably wouldn't have changed but the date would have been -- gone further and further out.
Q. Okay.
A. If that makes sense.
And I apologize. I just -- you know, like I said, we don't seem to have the whole thing. If we had the whole document, it would probably help me give a little more context of what the particular situation was.

Q. (BY MR. TETER) Does plaintiff's Exhibit 103 indicate to you that TriFlex/PCI in 1994 was unable to run Memory Read Multiple?
A. I don't believe --

MR. BRODY: Object to the form.
A. I'm sorry.
MR. BRODY: I'm sorry. I just wanted to object to form. Go ahead.

5 A. So I -- I don't believe that it -- it indicates
6 that PCI Memory Read Multiple did not run in 1994.

351. Based on all of the evidence cited above, OPTi's counsel appears to be misinterpreting the document. The document does not state that the MRM presnooping and prefetching feature did not work in February 1994.

10. Compaq conceived and reduced to practice over a lengthy period of time because it was making a very complex design, much more sophisticated than the relatively simple presnoop-only method described in the OPTi patent.

352. As discussed above, the OPTi presnooping method was much less sophisticated than the Compaq system or the Intel streaming method, and achieved results that fell far short of the Intel and Compaq chipsets. The OPTi presnoop patent, even as described in July 1995, failed to send data on every PCI clock, and as a result, was approximately half as fast as the Compaq CMC/DMC chipset and the Intel Triton. In view of the great complexity of the CMC/DMC, it is entirely consistent for Compaq to have conceived in late 1992, received silicon in February 1994, and released product in November 1994.

353. The deficiency in the OPTi design – its failure to send data on every clock cycle – suggests that the OPTi design was put together in a rush to compete with Intel and Compaq. The notion that OPTi would have conceived prior to Intel and Compaq, yet failed to figure out how to send data every clock cycle, is not plausible.

354. In October, 1994, Compaq had constructively reduced to practice an invention that included presnooping and prefetching, as set forth in Compaq's patent—Compaq filed the patent application that day, and I am informed that a patent application is a constructive reduction to practice of the methods and apparatus described therein.

355. Compaq/HP has provided clear and convincing evidence that it had reduced the presnooping feature in the CMC/DMC to practice by Februrary 1994, when Compaq/HP had working silicon and was in the debugging process.

356. Compaq/HP has provided clear and convincing evidence that the conception of the Triflex/PCI prefetching/presnooping invention had occurred prior to September 1993, the date of the Biner chipset drawings, and prior to August 1993, the date of the feature comparison. [EX. 188].

357. The Compaq inventors indicated that they worked continuously on the Triflex/PCI until it was reduced to practice in silicon, and as well in the Compaq patent application.

358. OPTi has not provided any evidence demonstrating that it conceived of the presnooping invention before the Compaq inventors. Rather, OPTi appears to have had great difficulty implementing the invention, and as late as July 1995 had not yet managed to make a PCI chipset that could send a data unit on every clock cycle. As a result, the OPTi invention trailed the conception and reduction to practice of the Mr. Thome and Mr. Collins in Houston, Texas.

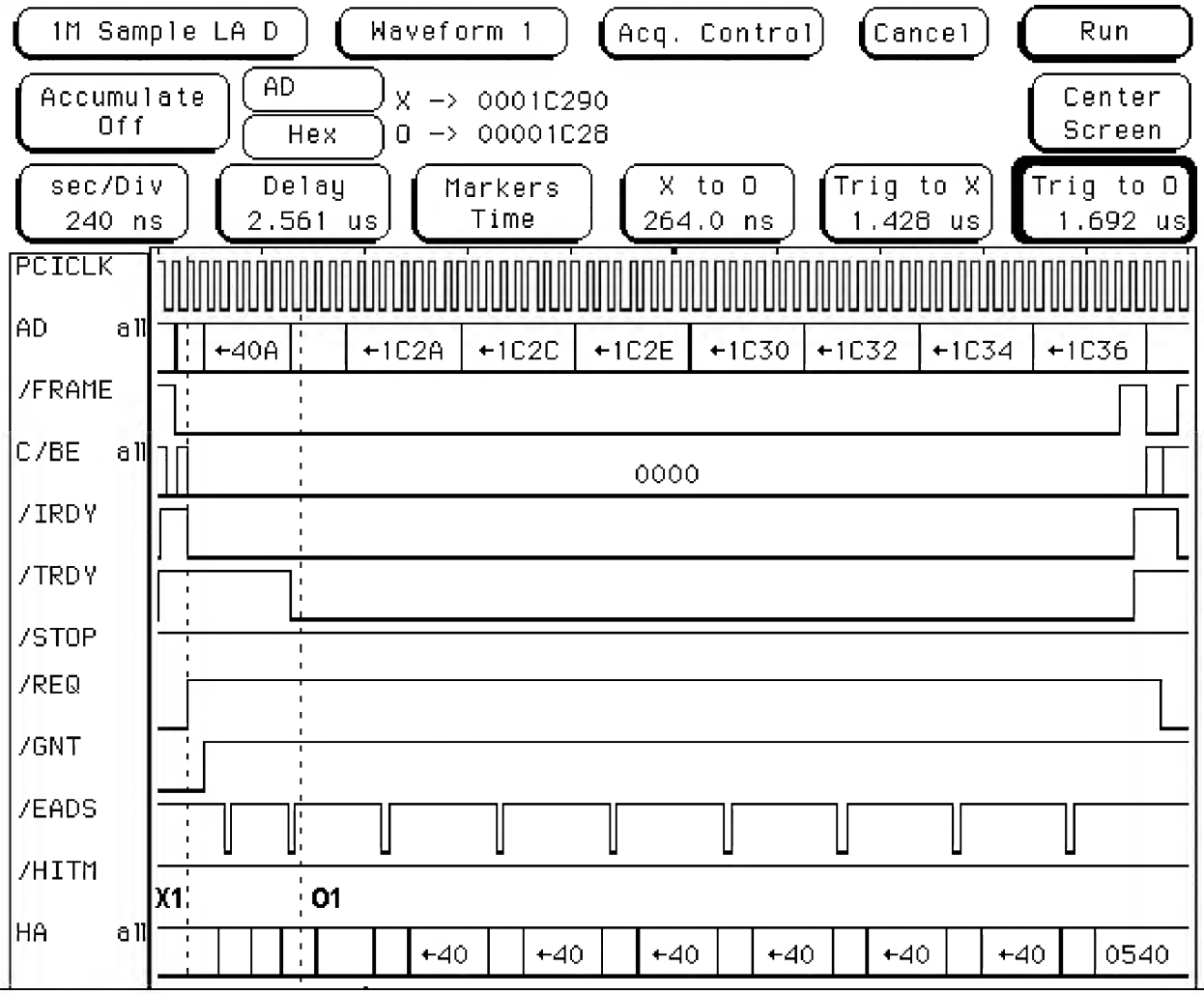
11. The Kramer Report shows that the Prosignia operated as the Compaq/HP witnesses testified.

359. The Compaq/HP evidence indicated that the Prosigna 500 included the presnooping and prefetching, allowing for a burst across cache line boundaries. In addition to evidence from Mr. Collins, Mr. Thome, and HP/Compaq itself, I have reviewed Silicon Valley Expert Witness Group (SVEWG's) Engineering report on the Compaq chipset ("Kramer report"). *See* Attachment C. The Kramer report shows that the Compaq chipset allowed streaming across cache line boundaries. Significantly, the Kramer report shows that the Compaq chipset implemented streaming across cache line boundaries by keeping the signals IRDY (initiator ready), TRDY (target ready), and FRAME asserted throughout the burst transaction. *See* Attachment C.

360. In addition, the Kramer report shows that the Prosignia 500 snooped only one line ahead, as recited in FIG. 9 of the '906 patent. Thus, the trigger for the next line

inquiry in the Prosignia 500 is clearly linked to the start of the transfer of the current line on the PCI bus. Figure C4 from the Kramer report, Attachment C, follows:

Figure C4. An example of a PCI master-initiated burst read transfer where the first inquiry cycle discovers the desired line is not cached cache modified in L1 cache.



The data being transferred over PCI (from the test buffer in main memory) set equal to its corresponding cache tag address (HA[17:5]) plus one 0 bit (corresponding to HA[4]) appended to the least significant end.

This clarifies the temporal relationship between the cache line tag addresses being snooped (with EADS# and HA) with the data in that cache lines that are being transferred over the PCI bus (AD). Note that the addresses being snooped (on HA at the time of the EADS# assertion) are not visible in this diagram because of its time scale. The values ending in "40" are *not* the addresses. Refer to Figures C4a-d to see the snoop address values displayed on the timing analysis.

(X1) is the address phase. This transfer is a PCI bus mastered Memory Read Multiple burst read from address 0x0001C290. (O1) is the first data phase (the first dword transfer).
[C4.bmp]

361. I should note that Mr. Kramer's analysis, though of a Prosignia 500, addresses a computer that had a Intel Pentium CPU at 150MHz, and that the BIOS had a date of July 1996. I understand that NVIDIA and Mr. Kramer have to date been unable to locate a Prosignia 500 with an earlier BIOS, and reserve the right to supplement this report in the event that Mr. Kramer and NVIDIA locate another Prosignia 500. In any event, I have no reason to believe that the presnooping and prefetching of the Prosignia 500 released in November 1994 would be any different than the presnooping and prefetching of the machine that Mr. Kramer tested and analyzed.

12. Acacia's analysis of the Compaq prior art is consistent with the rest of the evidence.

362. I have reviewed the transcript of Mr. Harris of Acacia. I understand that Acacia was considering buying the OPTi presnoop patents, and did not because Acacia identified the prior art Compaq patent. Mr. Harris testified about the discovery of the Compaq patent as follows:

31

Q And did you and Mr. Roop ever talk about this Compaq patent?
A Yes, we did.
Q And what did you guys talk about?
MR. GRACEY: Object to form.
THE WITNESS: I asked Mr. Roop what his thoughts were

32

regarding potential invalidity.
BY MR. BRIGHAM:
Q And what were his thoughts?
MR. GRACEY: Object to form.
THE WITNESS: He thought that we would have a problem if we tried to license the patents with regard to validity.
BY MR. BRIGHAM:
Q Because of the Compaq patent?
A Correct.
MR. GRACEY: Object to form.
BY MR. BRIGHAM:
Q And do you recall what you said?
MR. GRACEY: Object to form.
THE WITNESS: No, I don't recall.
BY MR. BRIGHAM:

16 Q Do you recall what your reaction was?
17 MR. GRACEY: Object to form.
18 THE WITNESS: Not right then.
19 BY MR. BRIGHAM:
20 Q You don't recall what your reaction was then?
21 A We decided, once we reviewed the patent, not to
22 make another offer.
23 Q So, did anything happened next with respect to
24 OPTi?
25 A I believe we pretty much stopped spending money

33

1 and time on that opportunity.

101

5 Q So, the only remaining factor was the validity
6 analysis; is that right?
7 A That was the issue that made us to decide not to
8 spend any more time with it.
9 Q Right.
10 But that was the issue that whether you went
11 forward with a higher offer or not hinged upon or whether you
12 went directly to the shareholders; is that right?
13 A For us, that was the showstopper.

363. Although I do not rely on the Acacia analysis per se, I agree with Acacia's conclusion. The Compaq prior invention is clear and convincing evidence that the asserted presnoop claims are invalid.

13. Compaq claim chart & timeline.

364. A claim chart is attached hereto as Attachment F, illustrating that Compaq's prior invention renders the presnoop patent claims-in-suit invalid. A timeline is attached hereto as Attachment G.

C. The asserted claims of the OPTi patents are invalid in view of the prior invention by Motorola inventors of the Motorola MPC105, and the MPC105 articles.

365. Between 1992 and 1994, Motorola conceived and designed a PowerPC-to-PCI bridge/memory controller chip called the Motorola MPC105. The MPC105 served as an interface between a PCI bus and system memory, and implemented both presnooping and prefetching features to facilitate a PCI bus master burst read of system memory. Motorola

conceived and implemented presnooping as described and claimed by the Patents-in-Suit by at least March of 1994.

1. The design of the MPC105 began in late 1992 or early 1993.

2. The design of the MPC105 had been completed by at least March of 1994 when Motorola began submitting papers describing the MPC105 to academic and commercial conferences.

372. Mr. Bryant's testimony is consistent with the documentary evidence produced by Alan J. Smith. Alan J. Smith, who has been retained by OPTi in this case, was an organizer of the Hot Chips VI Symposium in August of 1994. [*See* SMITH000047-000049]. Motorola presented the MPC105 at the Hot Chips VI Symposium (*see* EX. 66, NV020000-020007), and the organizers of the Symposium had placed an advertisement in the February 1994 *Newsletter of the Parallel Processing Technical Committee* asking for the submission of papers by March 24, 1994.

> **Hot Chips VI, August 14-16, 1994, Stanford University, Stanford, California, USA. Send an extended abstract (1-3 pages) by March 24, 1994 to Prof. Alan Jay Smith, Computer Science Division, UC Berkeley, Berkeley, CA 94720, Fax: (510)-339-6360, E-mail: smith@cs.berkeley.edu. Submissions can be made by hard copy, fax, or electronic mail (E-mail is preferred!).**
>
> [EX. 67 at NV 019294 (Newsletter of the Parallel Processing Technical Committee at 12)].

373. Mr. Bryant recalled that a presentation about the MPC105 was made at "Hot Chips" and believed that the MPC105 had "taped out" by the deadline for submission of papers:

31

16 Q. Were you ever the author of any articles
17 describing MPC105?
18 A. I helped with authoring articles, yes.

19 Q. Do you know if any of the articles were ever
20 present at any conferences?
21 A. Yes.
22 Q. Do you know whether there were slides associated
23 with presentations at conferences?
24 A. Uh-huh. Hot Chips.

35

3 Q. Do you have any reason to believe that an
4 extended abstract was not sent to Professor Alan Jay
5 Smith by March 24, 1994?
6 A. No.
7 Q. Do you believe that the MPC105 had taped-out by
8 March 24, 1994?
9 A. Yeah.

374. Alan J. Smith produced documents which confirm that another designer on the MPC105, Karl Wang, did indeed submit a paper describing the presnoop and prefetch features of the MPC105 by March 24, 1994:

> I am sending you by this e-mail an abstract of our paper we like to submit to the Hot Chips
> Symposium VI. This abstract contains slight modification from the abstract which I send
> to you by fax on March 24. Please accept this modified version. My appology
>
> [EX. 81 at SMITH000023 (3/25/94 Email from Karl Wang to Alan J. Smith)].

375. The paper that Karl Wang submitted to Alan J. Smith contained a description of the presnooping and prefetching features implemented in the MPC105:

> forwarded to memory. Finally,
> EAGLE implements a 32-byte PCI-to-memory read prefetch buffer. If speculative reads are
> enabled, at the completion of a PCI read EAGLE prefetchs the next cache line from memory to
> enhance the PCI-to-memory read performance. Since the processor, secondary
>
> [EX. 81 at SMITH000024 (3/25/94 Email from Karl Wang to Alan J. Smith)].

376. Moreover, the mail from Karl Wang to Alan J. Smith specified the status of the project as "first silicon":

> Status of Product: "first silicon" Product not yet announced. Please keep submission confidential.
>
> [EX. 81 at SMITH000028 (3/25/94 Email from Karl Wang to Alan J. Smith)].

377. Mr. Bryant confirmed that this sequence of events was consistent with his recollection:

36

Q. I would like to mark as Exhibit 81 a document labeled Smith 000023 to 000028.
(Defense Exhibit No. 81 marked.)
Q. Mr. Bryant, have you seen this document before?
A. No.
Q. Do you know who Karl Wang is?
A. Yes.
Q. Who is Karl Wang?
A. Karl Wang was the Central Control Unit team lead of the MPC105.
Q. What were his responsibilities?
A. Overseeing the core of the chip design.
Q. Do you remember him sending this e-mail for referencing its sending to Professor Alan J. Smith in March of 1994?
A. No.
Q. Do you remember contributing to an abstract that was sent to Professor Alan J. Smith?
A. I remember Karl asking me questions about how PCI worked. I didn't know what he wanted it for.
Q. Did you write any part of the abstract?
A. Not that I recall.
Q. If you look at Page 2 of Exhibit 81, this is a

37

document labeled Smith 000024. Down near the bottom, second to last sentence reads, "If speculative reads are enabled at the completion of the PCI read EAGLE prefetches the next cache line from memory to enhance the PCI to memory read performance." Do you see where I am referring to?
A. Yes, I do.
Q. What does "EAGLE" refer to?
A. EAGLE refers to our code name for the MPC105.
Q. And the time this e-mail was written on or around March 26th of 1994, was that an accurate description of how the MPC105 code named EAGLE behaved?
A. Yes.

3. Motorola received a working silicon chip for the MPC105 on May 22, 1994.

4. After reducing the MPC105 to silicon, Motorola continued to publish articles and prepared to sell the MPC105 commercially.

379. Motorola also participated in and presented the MPC105 at the ICCD 94 International Conference on Computer Design in October of 1994. [*See* EX. 61 at OPTINVIDIA 000681 (IEEE International Conference on Computer Design: VLSI in Computer and Processors, *Proceedings* at 412)]. The documentary evidence shows that the deadline for submission of papers to this conference was March 4, 1994, with a final manuscript due by July 1, 1994.

CALL FOR PAPERS

ICCD 94 International Conference on Computer Design, October 10-12, 1994, Royal Sonesta Hotel, Cambridge, Massachusetts. Original papers regarding work in the following areas are being solicited:

- Embedded systems
- VLSI & technology
- Architecture and algorithms
- Design & test
- Computer-aided design

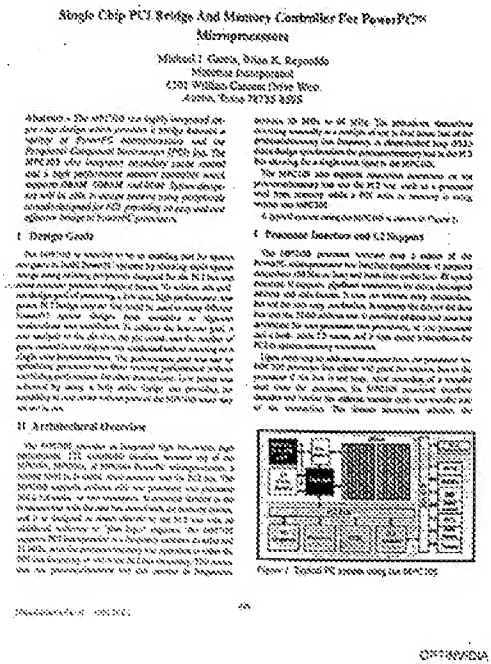
Submissions should be sent to technical program chair A.P. Ambler, Dept. Electrical Engineering & Electronics, Brunel University, Uxbridge, Middlesex, UB8 3PH, +44 895-203380, fax +44 895-258728, tony.ambler@brunel.ac.uk. Deadline for submissions is March 4, 1994, with notification of acceptance by April 30 and final manuscript due July 1.

[EX. 62 at NV 019301 (Spring 1994 *Design & Test of Computers* at 2)].

380. The paper Motorola presented at the conference (*see* EX. 61 at OPTINVIDIA 000676-000681) also clearly described the MPC105's presnooping and prefetching features.

...abled. The MPC105 also employs a speculative read feature which, if enabled, will snoop the next sequential cache line address when the current PCI read is accessing the third doubleword (8 bytes) of the cache line. Once the speculative snoop response is known and the read has finished, the data at the speculative address will be fetched and loaded into the buffer in anticipation of the same PCI master requesting it. If it is not requested, the data is just invalidated. In a typical system, the speculative feature will allow streaming of large amounts of data with very minimal delay on the PCI bus enabling the system to utilize the full bandwidth of the PCI bus.

[Ex. 61 at OPTINVIDIA 000681 (Proceedings, IEEE International Conference on Computer Design: VLSI in Computer and Processors at 412)].



381. The testimony of Mr. Bryant and Mr. Garcia, along with the calls for papers and the documentary evidence produced by Alan J. Smith, prove that the MPC105 design was completed and "taped out" in March 1994 at least, and that the design team had begun submitting papers describing the features of the MPC105 at that time. This evidence establishes that the MPC105 design team had conceived of presnooping and prefetching by at least March 24, 1994.

5. Motorola presented the MPC105 at Hot Chips VI in mid-August 1994 and formerly announced the MPC105 on August 22, 1994.

382. As Mr. Bryant testified, Motorola presented the MPC105 at the Hot Chips VI Symposium at Stanford University on August 14-16, 1994. [*See* SMITH000047-000269]. The testimony of Brian Reynolds confirms Mr. Bryant's recollection that the MPC105 was presented at Hot Chips:

9

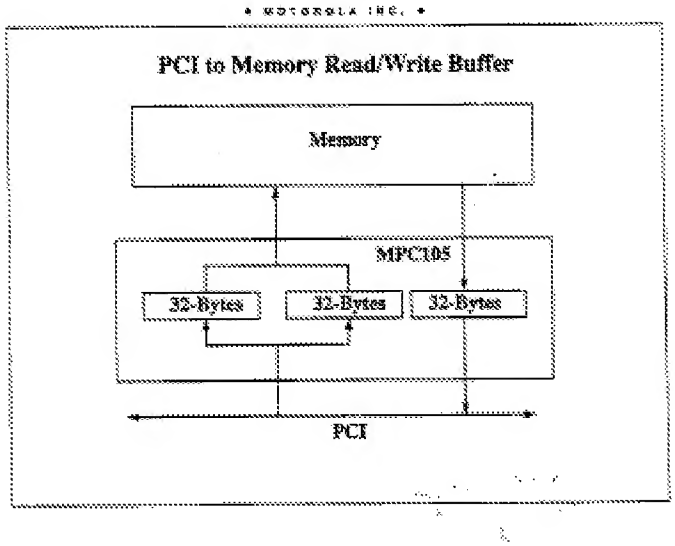
10 Q. Have you heard of the Hot Chips VI symposium
11 at Stanford?
12 A. Yes.
13 Q. And what was the Hot Chips VI symposium at
14 Stanford?

A. It's a technical design conference that people throughout the chip/semiconductor industry would attend and present some of their latest works.
Q. Did you present anything at the Hot Chips VI symposium at Stanford University in August of 1994?
A. No.
Q. Did any of your colleagues?
A. I believe Karl Wang presented. He was one of the design managers on the project.
Q. Do you know what Karl Wang presented in August '94 at the Hot Chips VI symposium at Stanford?

10

A. Not for sure, but I believe it was a presentation based on a paper that had been written about the MPC105.

383. In addition, Alan J. Smith has produced Motorola's presentation at the Hot Chips VI Symposium. [*See* SMITH000167-000174]. The presentation details the MPC105 features and explicitly shows the MPC105 32-byte PCI-to-Memory read prefetch buffer:

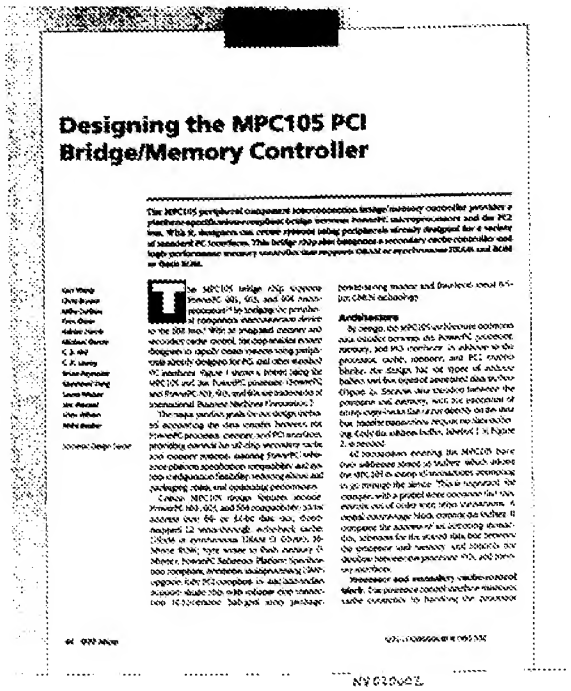


[SMITH000171 (Hot Chips VI Motorola Presentation at 5.2.5)].

384. The presentation that Motorola inventor Karl Wang presented at the Hot Chips IV Symposium, and the subsequent paper he published regarding it, describes the MPC105 presnoop and prefetch feature in accordance with the testimony and documentary evidence that I have cited in this report:

The MPC105 also implements a speculative read feature that, if enabled, will snoop the next sequential cache line address when the current PCI read accesses the third doubleword of the cache line. Once the snoop response is known and the read has finished, the device will fetch the data at the speculative address and load it into the buffer in anticipation of the same PCI master requesting it. In a typical system, the speculative feature will allow streaming of large amounts of data with very minimal delay on the PCI bus, enabling the system to use the full bandwidth of the PCI bus.

[EX. 66 at NV020005 (Hot Chips IV Symposium Paper at 47)].



Designing the MPC105 PCI Bridge/Memory Controller

385. In the summer of 1994, Motorola announced the MPC105 PowerPC-to-PCI bridge/memory controller. An article published by Electronic News on August 22, 1994 stated that the Motorola MPC105, code-named "Eagle", would be incorporated into a Motorola motherboard supporting PowerPC 603 and 604 processors. The announcement specifically referenced the presentation of the MPC105 at the Hot Chips VI Symposium, and stated that evaluation boards were then available, with manufacturing kits to be available in October of 1994:

StartArticles > Electronic News > August 22, 1994 > Article > Print Friendly

Motorola PowerPC move signals rivalry with IBM - Motorola rolls out reference motherboard, MPC105 bridge/memory controller - Product Announcement

Reinhardt Krause

AUSTIN, TEX.--Marking the point where it will seriously begin competing for customers with IBM, Motorola's RISC Microprocessor division last week began offering its own PowerPC reference motherboard and developer's program--as well as a new single-chip PowerPC-to-PCI bridge/memory controller.

While Motorola's reference motherboard is compliant with IBM's reference platform specification there are some differentiators. One is the new PowerPC-to-PCI bridge/memory controller--the MPC105 code-named Eagle--which was described at the Hot Chips VI symposium at Stanford University in Stanford, Calif. last week (see related stories, page 2). While immediate comparisons may be with Intel's Peripheral Component Interconnect (PCI) chipsets, the new Eagle bridge/memory controller also will compete with IBM's two-chip implementation, the 82650, unveiled last year (EN, Nov. 1, 1993).

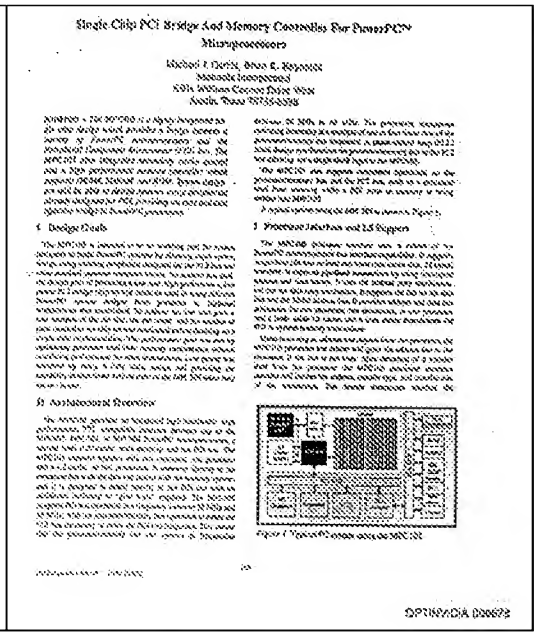
...

An evaluation kit is available now from Motorola for $25; a manufacturing kit will be shipping in October for $50. Motorola will initially be providing ARC (Advanced RISC Compliant) firmware and HAL (hardware abstraction layer) support and not the open-boot capability that IBM is still developing.

[EX. 59 at NV 012088-012089].

6. Motorola made additional presentations of the MPC105 in the latter half of 1994 and expected to begin offering the MPC105 for sale at the end of 1994.

386. As I cited above, Motorola again presented the MPC105 at the IEEE International Conference on Computer Design on October 10-12, 1994 and once again described the presnoop and prefetch features:

...aled. The MPC105 also employs a speculative read feature which, if enabled, will snoop the next sequential cache line address when the current PCI read is accessing the third doubleword (8 bytes) of the cache line. Once the speculative snoop response is known and the read has finished, the data at the speculative address will be fetched and loaded into the buffer in anticipation of the same PCI master requesting it. If it is not requested, the data is just invalidated. In a typical system, the speculative feature will allow streaming of large amounts of data with very minimal delay on the PCI bus enabling the system to utilize the full bandwidth of the PCI bus. [EX. 61 at OPTINVIDIA000676; OPTINVIDIA000681 (IEEE International Conference on Computer Design: VLSI in Computers and Processors, *Proceedings* at 412)].	

387. Mr. Reynolds testified that Motorola customers had operating boards prior to October 12, 1994:

149

Q. Okay. You said you thought that customers had operating boards at some point prior to October of

150

1994 -- or, I guess, October 21st of 1994.
A. I believe so, yes.
Q. Do you know how far in advance?
A. No.
Q. You said that the boards that were shipped were capable of performing the speculative read and speculative snoop functions as described. Did I hear that correctly?
A. Yes.

388. An article in Microprocessor Report on September 12, 1994, reported that Motorola expected shipments of the MPC105 to begin at the end of 1994:

Microprocessor Report; 9/12/1994; Gwennap, Linley

Newspapers

Aiming to simplify the design of PowerPC systems, Motorola and its partners have fired off a flurry of announcements. The most significant is that Motorola will market a system-logic chip, the MPC105, that will provide cache control, memory control, and a PCI interface for any PowerPC 60x processor. Although this chip is the second PowerPC system-logic design to be announced, it is the first with such a high degree of integration. Motorola expects to begin shipments by the end of this year.

[EX. 71 at NV20008].

7. The Motorola MPC105 was designed to operate as an interface between PowerPC 60x processors and a PCI bus.

389. Generally speaking, the Motorola MPC105 was a microchip interface between a PowerPC 60x CPU, a secondary (L2) cache, the PCI bus, and main memory. [*See* EX. 65 at NV 012239 (MPC105 User Manual at 1-1)]. The MPC105 connected to and communicated with each of these components and facilitated data transfers between them. The MPC105 is placed in context in Exhibit 63, which is an enlarged copy of Figure 1 from Motorola's publication at the IEEE Conference on Computer Design:

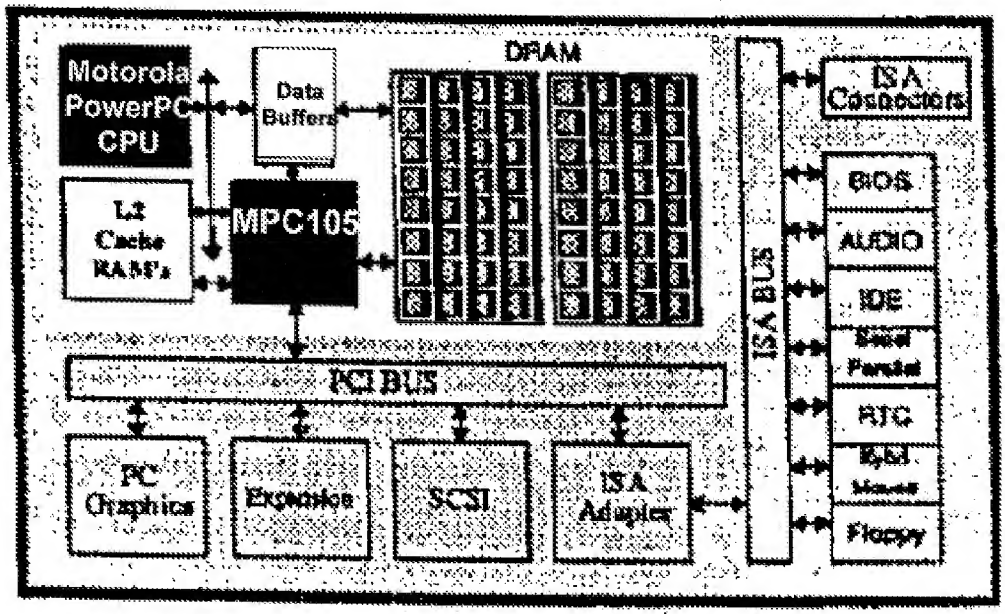
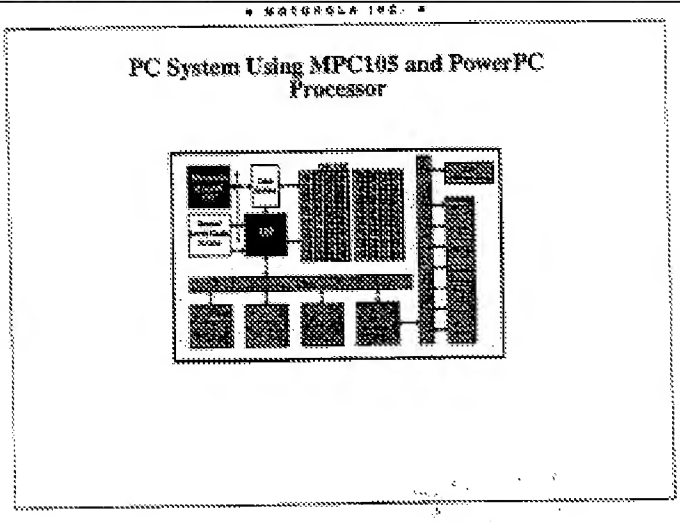


Figure 1 Typical PC system using the MPC105.

EXHIBIT 63

[EX. 63, (enlarged figure from Ex. 61, OPTINVIDIA 000678)].

390. The figure above was also displayed in Motorola's August 1994 presentation at the Hot Chips VI Symposium at Standard:



[SMITH00168 ((Hot Chips VI Motorola Presentation at 5.2.2)].

8. The MPC105 relied upon and implemented portions of the PCI Local Bus Specification, Revision 2.0, and also implemented presnooping as described in the Patents-in-Suit.

391. One of the primary purposes of the Motorola MPC105 was to facilitate data transfers between the PCI bus and main memory. [*See* EX. 65 at NV 012439-012459 (MPC105 User Manual, Ch. 7)]. In particular, the MPC105 user manual states that the MPC105 responded to PCI bus commands, including the PCI Memory Read Multiple command:

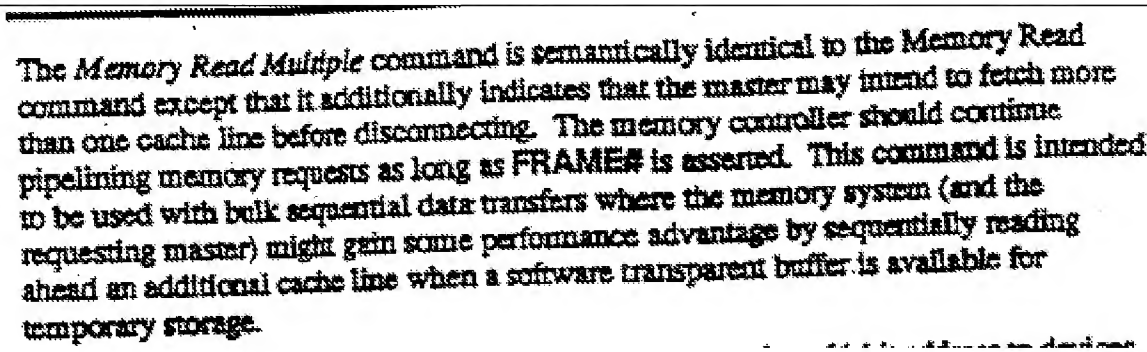
Table 7-1. PCI Bus Commands (Continued)

C/BE3–C/BE0	PCI Bus Command	MPC105 Supports as a Master Device	MPC105 Supports as a Target Device	Definition
1011	Configuration-write	Yes	No	The configuration-write command accesses the 256-byte configuration space of a PCI agent. A specific agent is selected when its IDSEL signal is asserted during the address phase. See Section 7.4.5.2, "Accessing the PCI Configuration Space," for more detail of configuration accesses.
1100	Memory-read-multiple	No	Yes	The memory-read-multiple command functions the same as the memory-read command on the MPC105. If prefetching is desired, speculative PCI reads should be enabled. See Section 8.1.3.2.1, "Speculative PCI Reads from System Memory," for more information.

[EX. 65 at NV 012444 (MPC105 User Manual at 7-6)].

392. As I have stated previously in this report, the PCI Memory Read Multiple command was designed as a hint that the bus master wishes to read more than one cache line in a transaction. The PCI Local Bus Specification, Revision 2.0 describes the Memory Read Multiple command as semantically identical to the PCI Memory Read command "except that it additionally indicates that the master may intend to fetch more than one cache line before disconnecting." In addition, the description of the PCI Memory Read Multiple command

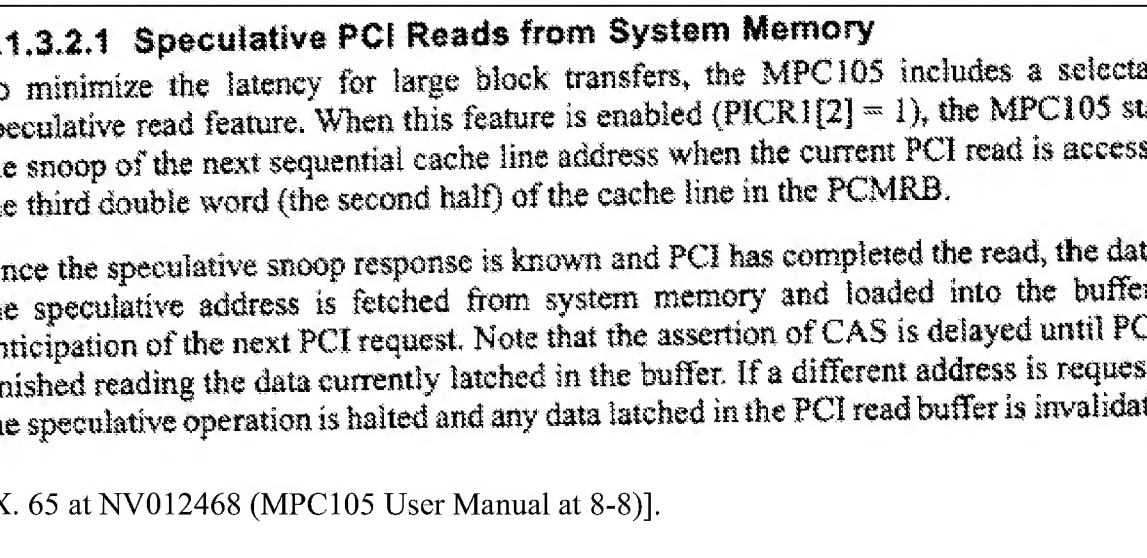
suggests that the memory system "might gain some performance advantage by sequentially reading ahead an additional cache line . . . .".



The *Memory Read Multiple* command is semantically identical to the Memory Read command except that it additionally indicates that the master may intend to fetch more than one cache line before disconnecting. The memory controller should continue pipelining memory requests as long as FRAME# is asserted. This command is intended to be used with bulk sequential data transfers where the memory system (and the requesting master) might gain some performance advantage by sequentially reading ahead an additional cache line when a software transparent buffer is available for temporary storage.

[EX. 64 at OPTINVIDIA 001546 (PCI Local Bus Specification, Revision 2.0 at 21)].

393. Indeed, the MPC105 responded to the PCI Memory Read Multiple Command by presnooping and prefetching. The MPC105 User Manual describes the presnoop as a "speculative snoop" and the prefetch as a "speculative read": "When this [speculative read] feature is enabled . . . the MPC105 starts the snoop of the next sequential cache line address when the current PCI read is accessing the third double word (the second half) of the cache line in the PCMRB [PCI-Read-from-System-Memory-Buffer]." [Ex. 65 at NV 012468 (MPC105 User Manual at 8-8)].



8.1.3.2.1 Speculative PCI Reads from System Memory

To minimize the latency for large block transfers, the MPC105 includes a selectable speculative read feature. When this feature is enabled (PICR1[2] = 1), the MPC105 starts the snoop of the next sequential cache line address when the current PCI read is accessing the third double word (the second half) of the cache line in the PCMRB.

Once the speculative snoop response is known and PCI has completed the read, the data at the speculative address is fetched from system memory and loaded into the buffer in anticipation of the next PCI request. Note that the assertion of CAS is delayed until PCI is finished reading the data currently latched in the buffer. If a different address is requested, the speculative operation is halted and any data latched in the PCI read buffer is invalidated.

[EX. 65 at NV012468 (MPC105 User Manual at 8-8)].

394. Each of the three different MPC105 designers that were deposed testified that the descriptions of presnooping and prefetching in the Hot Chips VI publication, the

IEEE publication, and the MPC105 User Manual were accurate and that the MPC105 operated as described. Michael Garcia testified that the MPC105 would snoop the next line while the current line was being transferred:

8

Q. Are you one of the designers of the Motorola MPC105 bridge chip?
A. Yes.
Q. Did the MPC105 bridge chip have a speculative snoop feature?
A. Yes.
Q. Did the MPC105 bridge chip have a speculative read feature?
A. Yes.

57

Q. And that system, while data was being transferred from a current line, would speculatively snoop the next line; is that right?
MR. BRODY: Object to the form.
THE WITNESS: Yes, if -- if it was to snoopable memory space.
Q. (BY MR. TETER) What is snoopable memory space?
A. PCI has two different types of memory space, some of it is snoopable and some of it is not, and you have to keep coherency in the part of the memory that's snoopable. And the part that's not snoopable, you don't care whether its coherent or not.
Q. So to follow that up, the system with the MPC described in Exhibit 61 would snoop snoopable memory for the next line while data units from the current line were being transferred to the PCI master; is that right?
A. Yes.

395. Brian Reynolds also testified that the MPC105 operated to presnoop and prefetch as described in the MPC105 User Manual:

7

Q. Are you one of the designers of the Motorola MPC105 bridge chip?
A. Yes.
Q. And at a high level did the MPC105 bridge chip have a speculative snoop feature?
A. Yes.
Q. Did the MPC105 bridge chip have a speculative read feature?
A. Yes.

72

Q. Looking at Exhibit 68, and the two papers that you published regarding the MPC105, did the MPC105 implemented in a computer system include a secondary memory?
A. Yes.

73

Q. A cache memory?
A. Yes.
Q. PCI master?
A. Yes.
Q. A PCI master that could sequentially read data units from secondary memory?
A. Yes.
Q. A snoop of the next line prior to the completion of the transfer of data units from the current line, did it include that?
A. Yes.

396. Christopher Bryant testified that he and the other designers implemented the Memory Read Multiple command in reliance upon its description in the PCI Local Bus Specification (EX. 64, Bryant Tr. at 16:3-19), and that the MPC105 consequently presnooped and prefetched in response:

21

Q. Did reading the PCI local bus specification motivate the MPC105 team to implement a presnoop or speculative read feature?
A. Yeah.
Q. What exactly motivated that implementation?
A. It is in the spec. You want to be PCI compliant then you're going to follow what's in the spec.

19

Q. What is the difference between the PCI memory read and the PCI memory read multiple commands?
A. There are none. The only difference is that it is a hint of what the bus master intends to do.
Q. And what is that hint?
A. That it is going to want more than one cache line of data.
Q. Did the MPC105 behave differently depending on whether it received a PCI memory read or a PCI memory read multiple command?
A. Yes. A memory read only fetched one cache of line of data. The memory read multiple, we ended up doing the speculative snoops and the speculative reads either from the L1 cache or the memory control.

9. The MPC105 had only one prefetch buffer, so it had to disconnect at cache line boundaries.

397. The MPC105 had two write prefetch buffers but only one read prefetch buffer. Mr. Garcia testified that Motorola considered the buffer size, but decided that increasing buffer size didn't add much performance:

18

Q. My question may have been bad. Let me just make sure I was right. So the MPC105 had two buffers for writes --
A. Correct.
Q. -- and then one buffer for reads?

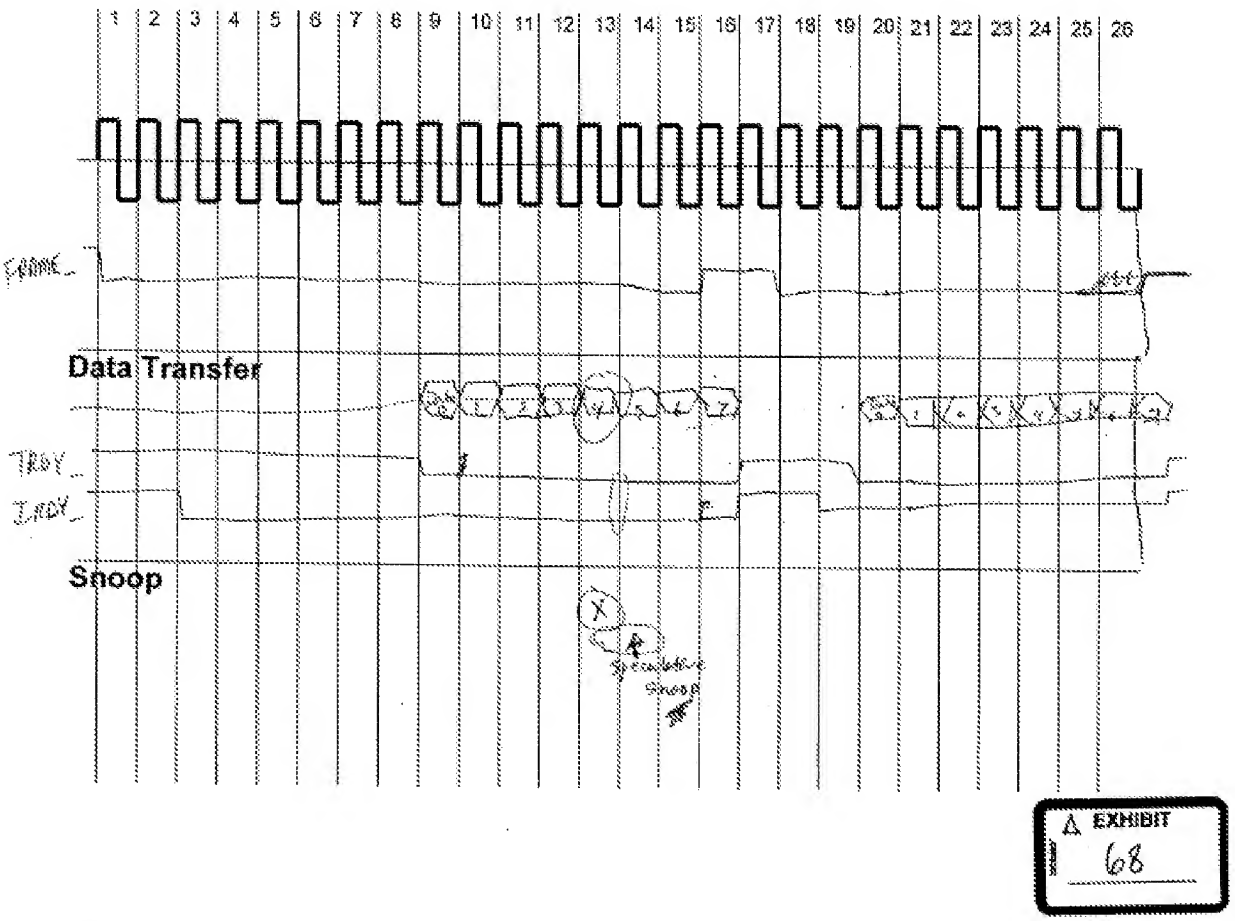
19

A. Right.
Q. Why did the MPC105 have only one buffer for reads and yet two buffers for write? Can you explain that?
A. It probably had to do with performance and trying to minimize the size of the chip and complexity. It could also mean that we didn't need it for performance-wise. It didn't buy us much to add an additional one.

19

The passage we just looked at indicates that with the two buffers for writes, one of the buffers could be being filled while the other buffer is forwarding to system memory. Is that how it worked?
A. Yes.

399. Thus, the MPC would disconnect at cache line boundaries. At his deposition Mr. Reynolds drew a timing diagram showing a next-line inquiry, the data transfers, and the gap (or disconnect) between the transfer of the data units of the first line and the data units of the next line.



EX. 68 (and associated Reynolds testimony).

400. Although the MPC105 disconnected at cache line boundaries, as Mr. Reynolds testified, MPC believed that the performance was sufficient and adding an additional prefetch buffer would not be necessary or worth the extra cost:

140

Q. When you were working on the MPC105, did you understand that the reason there would be the disconnect at a cache line was because you only had a buffer that was one cache line long?
A. Yes. You might say that it was also because -- yeah. We didn't -- and also because we wouldn't partially fill it with the next line was the other reason we had to target disconnect.
Q. If you understood that the target disconnect was because you had a buffer that was one cache line long, why, then, didn't you just have two PCMRB buffers?

A. I think it was because the additional area and complexity of controlling the second buffer was more than we wanted to do. We also believed that the

141

performance benefit of the speculative reads was sufficient for the trade-off of -- of the area for a second one. It starts to -- I guess even with one it looks like a cache. The more you have, the more states -- or more you have to -- you have to track for each individual one that stayed in the system. I was going to say something else.

38

Q. Turning to the page prior, 411, in your paper --

A. Uh-huh.

Q. -- the last paragraph, we looked at it before, states, "There are three PCI" --

A. Right.

Q. -- "to system memory transaction buffers, one for reads and two for writes." Do you see that?

A. Yes.

Q. Is that the way the MPC105 was actually implemented?

A. Yes.

Q. Why did you design the system to have two buffers for PCI writes to system memory, and one buffer

39

for PCI reads from system memory?

A. We had two buffers for PCI writes to system memory so that we could post or store data that's being written by PCI and not have to flush that buffer to the main memory immediately if there is a subsequent write on the PCI bus to use it. So the main goal was to allow the maximum amount of data bandwidth on the PCI bus.

And with the writes we determined that we -- by the time the first write had written a full cache line, we weren't really guaranteed that we could have gotten the snoop done for that write and the data pushed out of the buffer and into main memory. So we had a second buffer to take up the slack, so while the first one is doing its snoop and pushing data to memory, the second one is being filled. And then it would switch while the second one is being filled -- or, sorry, is doing its snoop and writing to memory the first one would be filled.

On the read side, the -- because we were doing the speculative snooping, we only felt we needed one buffer -- the buffers take up area and add more complexity to the chip, more possibility of errors.

But, anyway, the read one we determined -- had the same goal, was to optimize the amount of bandwidth that you

40

could use on the PCI bus for data. And so we -- I don't know why we didn't particularly use a second buffer. We probably could have.

But it starts to look like another cache, basically, because we were more storing more and more lines of data as the owner. And so -- but mostly was -- we decided the performance was sufficient to only use the one buffer, use the speculative snooping and prefetch feature, to get the second line of data in without a significant loss in bandwidth on the PCI bus.

401. In addition, Mr. Reynolds testified, Motorola believed that there simply weren't bus master devices at the time that would use the MRM command.

144

Q. So was it -- was it your view at the time that the speculative read feature allowed transactions to be fast enough?

A. Yes. And at the time we also were not convinced for the 105 that there were devices that were using memory read multiple as a command. There probably were, but it was not -- I don't think it was prevalent. And so that also factored into our decision to --

Q. Are you suggesting that at the time there wasn't as much of a need for a bridge chip that would allow for memory read multiple commands without any kind of latency at all between data transfers?

A. Yes. Yes.

402. The motivation to modify the MPC105 to include a buffer that could avoid the disconnect could have come from any number of sources, including the PCI Design Guide EX. 97, which suggested prefetching (see discussion above in section re: Intel) or from the Amini patent, which taught the "ping-pong" structure of prefetch buffers. Using a "ping-pong" buffer structure, as taught by Amini, would have allowed the MPC105 to avoid any disconnect. In view of the cost/benefit considerations, the MPC105 inventors simply concluded that one buffer was enough for their purposes.

10. The MPC105's disconnect at cache line boundaries is not relevant to several of the '906 claims, and even under OPTi's interpretation, '906 claims 9 and 26 would be obvious in view of the MPC105 system.

403. '906 patent claims 1, 7, and 8 do not exclude a sequential transfer of data units where the PCI master disconnects at cache line boundaries, as in the MPC105. '906 patent claim 9 states that the transfers must be "at a constant rate," which OPTi apparently states would prohibit the PCI master from disconnecting at a cache line boundary, or otherwise preclude additional wait states between cache lines. "Constant rate" does not preclude wait states, as OPTi's own '906 patent does not transfer data on every PCI clock cycle. In any event, assuming that OPTi's construction is correct, and that no additional wait states can appear at a cache line boundary, claims 9 and 26 would be obvious – Motorola did not include another cache line buffer in the MPC105 bridge simply because it believed that the performance of the MPC105 was sufficient, and adding another buffer (which would allow a transfer at a constant rate in the MPC105 system) would have been obvious.

11. The MPC105, as described by the evidence, was faster than the Viper described in the OPTi '906 patent.

404. During the MPC105 depositions, OPTi counsel asked whether the MPC105 inventors would have wanted the MPC105 to transfer data faster in MRM transactions. According to one of the Motorola papers, the MPC105 could transfer data in an MRM in the following cycles:

Table A. Key system performance assuming 66-MHz processor/memory bus, 33-MHz PCI bus, 60-ns DRAM, and 9-ns synchronous-burst SRAM.

Transaction	Number of cycles
Processor-to-memory	
Burst read from L2	3-1-1-1
Pipelined burst-read from L2	3-1-1-1/1-1-1-1
Burst write to L2	3-1-1-1
Burst read from DRAM (SDRAM)	8-3-3-3 (8-1-1-1)
Burst write to DRAM (SDRAM)	8-3-3-3 (5-1-1-1)
PCI-to-memory	
Pipelined burst-read	9-1-1-1-1-1-1-1/ 4-1-1-1-1-1-1-1
Pipelined burst-write	2-1-1-1-1-1-1-1/ 2-1-1-1-1-1-1-1
Processor-to-PCI	
Single read	12
Single write	5

[EX. 66 at NV020006].

405. Using the Motorola paper as an example, for a MRM transfer, each cache line after the first cache line would be transferred in 4-1-1-1-1-1-1-1, or 11 cycles per cache line.

406. For an MRM transfer in the OPTi Viper shown in the '906 patent, each cache line after the first cache line would be transferred in 16 cycles, because the OPTi Viper inserted a wait state between the transfer of each data unit on the PCI bus.

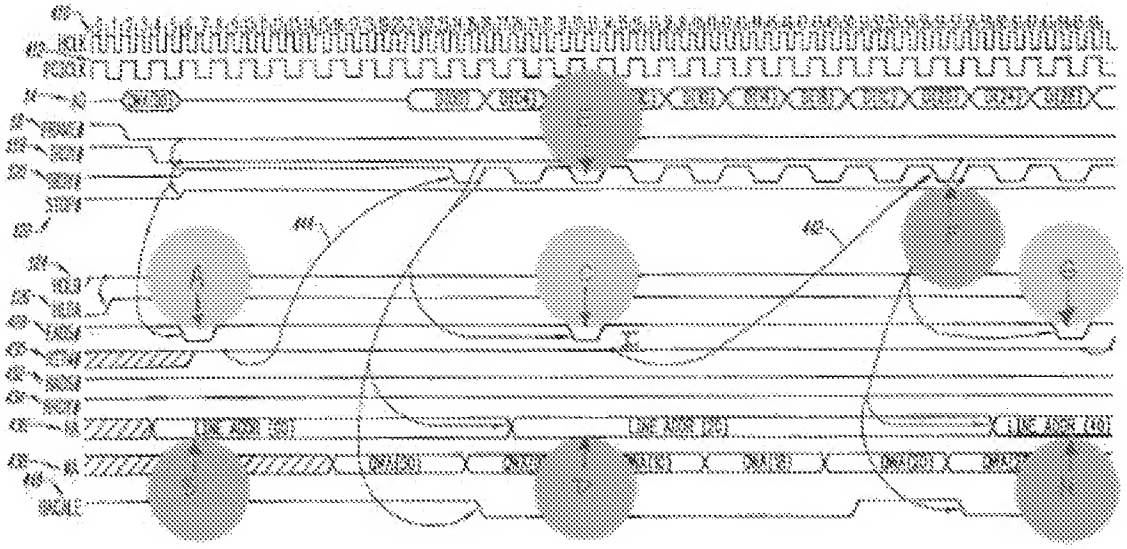


FIG. 4

12. Summary of opinions to be expressed with respect to the Motorola MPC105.

407. The testimony of Michael Garcia, Brian Reynolds, and Christopher Bryant, is consistent with the text of the MPC105 User Manual and the numerous publications and presentations about the MPC105 that were presented to academic and commercial organizations in 1994. The testimony and documentary evidence shows that the design of the MPC105, including the implementation of the presnoop and prefetch features, was completed by at least March of 1994.

13. Claim chart & timeline.

408. A claim chart is attached hereto as Attachment H, illustrating that Motorola's prior invention renders the presnoop patents claims-in-suit invalid for anticipation or obviousness. A timeline is attached hereto as Attachment I.

D. The IBM prior art (MicroChannel TDB and Amini).

1. IBM Microchannel TDB (1993).

409. Claims 1, 7, 8, 9, 21, and 26 of the '906 Patent are rendered obvious under § 103 by IBM's Micro Channel architecture described in the IBM TDB. As such, the asserted claims of the '906 Patent are invalid. The Micro Channel architecture includes a snoop of a second or next cache line during the transfer of the data in a first cache line, to reduce the latency caused during a burst transfer of data when the second cache line is not snooped until after the data in the first cache line is transferred. Thus, the architecture teaches predictive snooping.

410. IBM published a technical disclosure bulletin in October 1993, two years prior to OPTi's application for the '906 Patent. IBM's disclosure constitutes a 35 U.S.C. § 102(b) reference that cannot be sworn behind by OPTi. Since the disclosure renders obvious the subject matter of the '906 Patent, it renders the asserted claims of the patent invalid. A claim chart for the IBM Micro Channel art is attached hereto as Attachment J.

2. Amini (1993).

411. Claims 1, 7, 8, 9, 21, and 26 of the '906 Patent are anticipated under § 102 and/or rendered obvious under § 103 by the IBM Amini patent. U.S. Patent No. 5,581,714. (EX. 215).

412. IBM was one of the members of PCI SIG, and like fellow members Intel and Compaq, worked on a chipset to implement the PCI specification MRM command.

413. The IBM Amini patent describes a bridge chipset that includes a prefetch buffer that may be implemented in a "ping pong" structure so that data may be continuously transferred from the secondary memory to the bus master. Mr. Amini testified:

20

Q If you could look at Column 11 of Exhibit 215. Beginning at line 6, it reads:
"The FIFO may be implemented, for example, in a double word (DWORD) interleaved structure or a ping-pong structure."
Do you see that? 1:30:12PM
A Yes.
Q What is a ping-pong structure?
A I have -- I'd have to take a closer look specifically to refresh my memory. 1:30:28PM

21

Q Okay.
A But more than likely, at a high level, it probably references to being able to write into it from one bus and read from it to the other bus at the same time. 1:30:30PM 1:30:49PM
Q So would a ping-pong structure include two structures, one that could write while the other was reading at the same time?
MR. BRODY: Object to the form. 1:31:07PM
THE WITNESS: Can you ask your question again?
BY MR. TETER:
Q Sure.
Would a ping-pong FIFO include two components, one that could write while the other one was reading? 1:31:13PM
MR. BRODY: Object to the form.
THE WITNESS: Probably, yes.

414. With a buffer implemented in a ping-pong structure, the Amini patent would allow bursting across cache line boundaries without delay, as the buffer would start to fill from the DRAM as soon as data units left the buffer for the PCI bus master and space became available. Mr. Amini testified:

66

What's your understanding, if any, of what it means that data is transferred into the FIFO 2:47:51PM

67

immediately when space becomes available in the 2:47:54PM FIFO?

MR. BRODY: Object to the form; asked and answered.

THE WITNESS: As I read this, you know, my 2:48:01PM sense would be that by that, it meant that that location that's empty can be used to -- to transfer additional data into.

And that would be consistent with a concept of, you know, ping-pong or trying to improve 2:48:19PM the efficiency of the other transfers between two busses.

415. A claim chart for the Amini patent is attached hereto as Attachment K.

E. In addition to being anticipated, the claims in suit were obvious in view of the Intel (including Shanley), Compaq, Motorola and IBM prior art.

1. Differences between the invention and the prior art.

416. The obviousness instructions state that the differences between the prior art and the claimed invention should be considered as follows:

> 10.9.2 DIFFERENCES BETWEEN THE INVENTION OF THE CLAIMS AND THE PRIOR ART
>
> In determining the differences between the invention covered by the patent claims and the prior art, you should not look at the individual differences in isolation. You must consider the claimed invention as a whole and determine whether or not it would have been obvious in light of all of the prior art.
>
> In deciding whether to combine what is described in various items of prior art, you should keep in mind that there must be some motivation or suggestion for a skilled person to make the combination covered by the patent claims. You should also consider whether or not the prior art "teaches away" from the invention covered by the patent claims. The

question to be answered is: Would someone reading the prior art be discouraged from following the path taken by the inventor?

417. The prior art does not "teach away" from the OPTi presnoop invention. In fact, the prior art, beginning with the Shanley text, clearly encouraged persons of skill in the art to practice snoop ahead.

2. Level of ordinary skill.

418. The instructions address the level of skill in the art as follows:

10.9.3 LEVEL OF ORDINARY SKILL
Obviousness is determined from the perspective of a person of ordinary skill in the art. This person is presumed to know all of the prior art, not just what the inventor may have known. When faced with a problem, this ordinary skilled person is able to apply his or her experience and ability to the problem and also to look to any available prior art to help solve the problem.
Factors to consider in determining the level of ordinary skill in the art include the educational level and experience of people working in the field, the types of problems faced by workers in the art and the solutions found to those problems, and the sophistication of the technology in the field.

419. I addressed the level of ordinary skill in the art earlier in my report. It is my opinion that the level of ordinary skill in the field was relatively high, as persons in the chipset design field typically had practical work experience that would have led them to practice the claimed presnoop invention.

3. The objective indications of obviousness support my opinion.

420. As noted above, I have been given the following instruction concerning the law of obviousness:

10.9.4 OBJECTIVE INDICATIONS CONCERNING OBVIOUSNESS
You also must consider what are referred to as objective indications of non-obviousness. Some of these indications of non-obviousness are:
1. Commercial success of products covered by the patent claims or made by a process covered by the patent claims.
2. A long-felt need for the invention.
3. Failed attempts by others to make the invention.
4. Copying of the invention by others in the field.
5. Unexpected results achieved by the invention.
6. Praise of the invention by the infringer or others in the field.
7. The taking of licenses under the patent by others.

8. Expressions of surprise by experts and those skilled in the art at the making of the invention.
9. The patentee proceeded contrary to the accepted wisdom of the prior art.
The presence of any of these objective indications may suggest that the invention was not obvious. These objective indications are only relevant to obviousness if there is a connection, or nexus, between them and the invention covered by the patent claims. For example, commercial success is relevant to obviousness only if the success of the product is related to a feature of the patent claims. If the commercial success is the result of something else, such as innovative marketing, and not to a patented feature, then you should not consider it to be an indication of non-obviousness.

421. Although OPTi has not identified any evidence that it plans to argue what constitutes "objective indications of obviousness" I will analyze the factors below.

a. The OPTi Viper was not a commercial success—Intel's Triton, which included both prefetching and presnooping, had far greater performance and was successful.

422. OPTi's Viper was not a commercial success. An August 1995 article in the Microprocessor Report article explained:

Opti Fights for Market Share

Opti's long-awaited Viper-M chip set has been in production since May 1995, about three months behind Triton and Apollo Plus. As Figure 3 shows, Viper-M's system architecture is not as straightforward as that of other chip sets: part of the ISA bus connects to the system controller, and the rest connects to the peripheral controller.

In addition to the common features for a Pentium-class chip set, Viper-m has a few unique features. It supports the Sony Sonic-2WP cache module and adaptive write-back, which means that if a CPU write hits in the secondary cache and is a page hit in main memory, the write will be treated as a write-through cycle. Viper-M can also run its PCI interface in either synchronous or asynchronous modes. In addition, it is the only chip set discussed in this article that supports a VL-bus slave device. The VL support, which complicates the Viper-M architecture, really belongs to 486-class systems. It seems unlikely that this feature will help Opti gain market share.

Opti has been very successful in the chip-set business for many years. Its total market share for desktop PCs was about 20% in 1994. Like VLSI Technology, Opti is losing market share this year for three reasons. First, the Viper-M product has been late to market. Second, the product does not bring enough differentiation in performance and function. In particular, its PCI performance is relatively poor: the sustained data throughout is

> about 60 Mbytes/s, while all the other chip sets discussed in this article can sustain 100 Mbytes/s or more. Although Opti calls Viper-M a multimedia chip set, it does not have the features to separate it from others as a dedicated multimedia solution. Finally, Intel has simply taken away Opti's market share.
>
> Ironically, Opti has always been thought of as a low-cost chip maker, but it cannot compete with companies like VIA on cost. Opti was once the number-one chip-set supplier to Taiwan, but Opti's presence in the Far East market, which consumes more than 50% of all chip sets worldwide, is not significant any more. Opti seems unable to sustain its profit margins by selling to that market. In the U.S., Intel has stolen major system OEMs like Packard Bell, Gateway, and Dell that used to be either Opti or VLSI customers.

[Yong Yao, Vendors fight for Pentium core-logic market; Intel is king of the land, most others are losing chip-set share (Cover story), MICROPROCESSOR REPORT (August 21, 1995) (NV 021342-021348)].

423. Triton's commercial success was due to many factors – not just presnooping. As the Viper demonstrated, presnooping alone did not ensure success. Intel exceeded Viper's performance by including prefetch buffers that OPTi did not have.

b. No long-felt need existed – the PCI specification was very new and bus masters did not initially use the MRM command.

424. The presnoop invention was conceived by Intel, Compaq, Motorola, and OPTi at a time when the PCI specification was very new. As a result, those in the industry believed that few bus masters would be trying to take advantage of the MRM command. For example, Mr. Reynolds testified, Motorola believed that there simply weren't bus master devices at the time that would use the MRM command.

144

Q. So was it -- was it your view at the time that the speculative read feature allowed transactions to be fast enough?
A. Yes. And at the time we also were not convinced for the 105 that there were devices that were using memory read multiple as a command. There probably were, but it was not -- I don't think it was prevalent. And so that also factored into our decision to --
Q. Are you suggesting that at the time there wasn't as much of a need for a bridge chip that would

14 allow for memory read multiple commands without any
15 kind of latency at all between data transfers?
16 A. Yes. Yes.

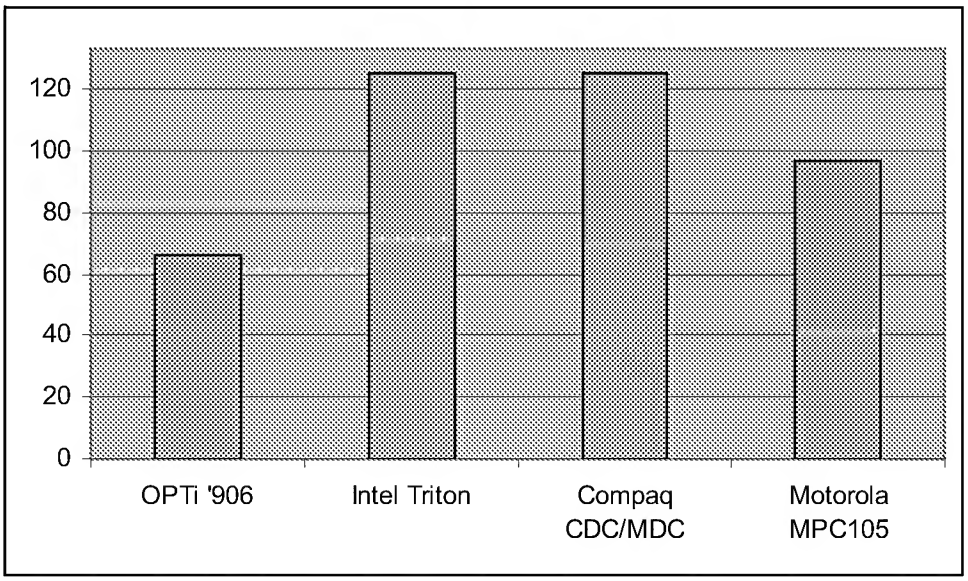
c. Others did not try and fail to make the invention.

425. Others in the field did not fail to make the invention. In fact, many of the competitors in the field implemented *both* presnooping and prefetching – it appears from OPTi's patent and the articles cited above that OPTi's Viper chipset implemented only presnooping, and as a result, had a wait state inserted between each data transfer and a very low data transfer rate.

d. No evidence suggests that anyone copied the Viper – OPTi trailed Intel and others in PCI chipset performance.

426. No evidence suggests that anyone copied OPTi. Rather, as described above in my report and in the cited article, OPTi was viewed as lagging in performance to Intel's Triton.

427. Because Intel's Triton and Compaq's CMC/DMC sent data units on each PCI clock cycle, they could achieve a 133 MB/s data transfer rate even across cache line boundaries. (The data rate for the first cache line was less.) According to the MPC105 articles, the MPC105, which could send DWORDs at a 4-1-1-1-1-1-1-1 clock cycles, should have been able to achieve a data rate of approximately 133*(8/11) MB/s, or 96 MB/s. According to the OPTi '906 patent, the system would have a wait state between each data transfer. Thus, according to the '906 patent figures, the '906 patent should have been able to achieve a data rate of only 133/2 or 67 MB/s. *See* Figure, below.



Performance Comparison for Steady-State MRM Transfer Across Cache Lines In MB/s

e. The invention did not achieve unexpected results.

428. Presnooping does not achieve unexpected results – the results are entirely predictable and obvious.

f. The invention did not receive praise of others.

429. OPTi's Viper was not praised, but was criticized by commentators as late arriving to market and as having poor performance, much slower than Intel's Triton. The OPTi inventor, Mr. Ghosh, testified that he did not publish any awards or articles describing the presnoop feature.

170

Q. Did you ever publish any articles about the pre-snoop feature that you describe and claim in the '906 patent?
A. No.

171

Q. Did you ever win any awards for the predictive snooping invention?
A. Awards from industry, you mean?
Q. Industry. Sure.
A. No.
Q. Did OPTi give you a bonus for the predictive snooping method?
A. Yes.

9 Q. What was the bonus?
10 A. Long back. Maybe 2,000, maybe $5,000.

430. Shanley and others did praise Intel's Triton – but the Triton succeeded and was praised for many reasons, not just presnoop. Triton achieved the performance gain by both presnooping and prefetching. The Viper, as described in the '906 patent, only presnooped, and its performance lagged far behind. Similarly, the Compaq CMC/DMC chipset also prefetched and presnooped, and achieved far greater performance than the '906 patent.

431. Intel and Compaq did file patent applications for the Intel streaming invention (combination of presnoop and prefetch elements) and the Compaq concurrency streaming invention (combination of presnoop, prefetch, and concurrency in other operations on other buses in the system), but those patent applications covered much more than the mere presnoop that OPTi has claimed in the '906 patent. Intel and Compaq each described and claimed a more complicated combination of elements that allowed Intel and Compaq to achieve the significant performance gains that they reported and attained. As a result, the fact that Intel and Compaq filed patent applications for their inventions does not show that OPTi's '906 presnoop claims were novel.

g. The taking of licenses under the patent by others.

432. I understand that AMD did not take a license from OPTi. VIA did not take a license. Apple did not take a license. I am informed that Compaq indicated, through Mr. Dukker, that it did not want a license. The only party licensed to the OPTi portfolio is Intel. I addressed the Intel agreement above, and have explained why it does not suggest that OPTi's presnoop patents are valid.

433. OPTi's decision not to sue VIA, which failed to take a license, is noteworthy. VIA indicated that its own prior art chipset, the VT82C505, practiced presnooping and would invalidate the OPTi presnoop patents. The parties exchanged timing diagrams and disputed whether the VIA chipset operated as VIA stated it did.

434. OPTi states that it found a VIA board and could not reproduce the tests that VIA performed. According to Mr. Marren, VIA was lying.

Mr. Ferrell indicated that he believed the wave forms would conclusively proved that Via was published and was practicing the '906 invention more that one year prior to '906 filing by OPTi. What did OPTi conclude after seeing the wave forms in Exhibit 260?
MR. GRACEY: You can answer that to the extent that was communicated to Via.
THE WITNESS: Yes. But we communicated to Via, it was total unadulterated bogus.

236

Q So was it your belief at that time that Via was lying to OPTi?
A Yes.

236

435. Mr. Marren testified that OPTi does not agree that the VIA prior art practiced presnoop, but his belief is based on a board that OPTi no longer has, and cannot find.

Q Has OPTi agreed now at this point that the Via prior art chipset in fact did practice presnoop?

243

MR. GRACEY: Object to form.
THE WITNESS: The answer to that is absolutely no.
BY MR. TETER:
Q And that's based on OPTi's own testing of the board that you described earlier?
MR. GRACEY: Object to form.
THE WITNESS: Yes.
Q
BY MR. TETER:
Q And that board no longer exists at OPTi, correct?
A We are not sure yet.
Q You're not sure?
A We are still looking.
Q The board may be at OPTi?
A No, it isn't at OPTi, but maybe somewhere we can locate it. We are still looking. We are doing a search to see if we can find that board or a duplicate of it.
Q Where are you looking for it?
A In Canada and in the U.S.

23 Q Did you ask Via to return it?
24 A I believe so.
25 Q What was Via's response?
244
1 A They would try.
2 Q They would try?
3 A They would try.
245

h. No experts expressed surprise at the making of the invention—presnoop was fully expected and inevitable.

436. I am not aware of any expression of surprise by anyone of skill in the art at the making of OPTi's Viper. As noted above, the Viper was criticized by commentators in the field.

437. The OPTi inventor, Mr. Ghosh, testified that nobody told him they were surprised by the results.

171
13 Q. Did anyone ever tell you that they didn't
14 think that the pre-snooping system and method could
15 be done?
16 A. I don't get your question.
17 Q. Did anyone ever say, Hey, it surprised me
18 that this worked. I didn't think it could be done?
19 A. I have never heard such a comment.

i. The patentee did not proceed contrary to the accepted wisdom of the prior art.

438. The conventional wisdom, as set forth in the Shanley text, was that presnooping was a technique that could improve performance. The earliest chipsets, such as Saturn, Mercury, and Neptune, did not include snoop ahead for the reasons discussed at length above. However, the "conventional wisdom" held that read ahead, and snoop ahead, should be used if practical. [*See* Shanley Text (EX. 101); Design Guide (EX. 97) (advocating read ahead); PCI Spec. 2.0 (EX. 64)].

F. OPTi's embodiment of the invention was a generation behind Intel and Compaq.

439. Consistent with the testimony of former OPTi founder Jaswa, I have noted above that OPTi appears to have been behind in PCI chipset development.

440. Unlike both the Intel Triton and Compaq Triflex/PCI chipsets, the OPTi Viper-M Chipset does not appear to utilize the peak 132 MB/s throughput available to PCI bus masters. The PCI local bus runs at 33 MHz and utilizes a 32-bit (or 4-byte) data path. As a result, the PCI local bus is capable of transferring 132 million bytes per second (33,000,000 cycles per second at 4 bytes per cycle) when bursting data. [*See also* EX. 64, PCI Local Bus Specification, Revision 2.0].

> **1.5. PCI Local Bus Features and Benefits**
>
> The PCI Local Bus was specified to establish a high performance local bus standard for several generations of product. The PCI specification provides a selection of features that can achieve multiple price-performance points and can enable functions that allow differentiation at the system and component level. Features are categorized by benefit below:
>
> | | |
> |---|---|
> | High Performance | • Transparent upgrade from 32-bit data path (132 MB/s peak) to 64-bit data path (264 MB/s peak). |

[EX. 64 at OPTINVIDIA 001530 (PCI Local Bus Specification, Revision 2.0 at 5)].

441. The evidence cited herein establishes that both the Intel Triton and the Compaq Triflex/PCI chipsets were capable of transferring data in every PCI clock period, continuously and across cache line boundaries. The OPTi Viper-M chipset, however, does not appear to have been capable of transferring data in every PCI clock period.

442. For example, OPTi's '906 Patent shows timing diagrams with intermittent wait states between data transfers of the PCI bus. Figure 4 of OPTi's '906 patent purports to show "the operation of the system of FIG. 1 in a situation where a PCI master has requested a burst read access to an address at the beginning of a cache line-sized block in the secondary memory address space . . . ." [EX. 17, '906 Patent Col. 12:21-25]. Figure 4 clearly

shows intermittent wait states between 32-bit data transfers on the PCI bus. These wait states are shown by the non-assertion (high) of the TRDY signal line in Figure 4. Figure 4 thus shows that each 32-bit data transfer on the PCI bus is interrupted by a wait state when TRDY is not asserted.

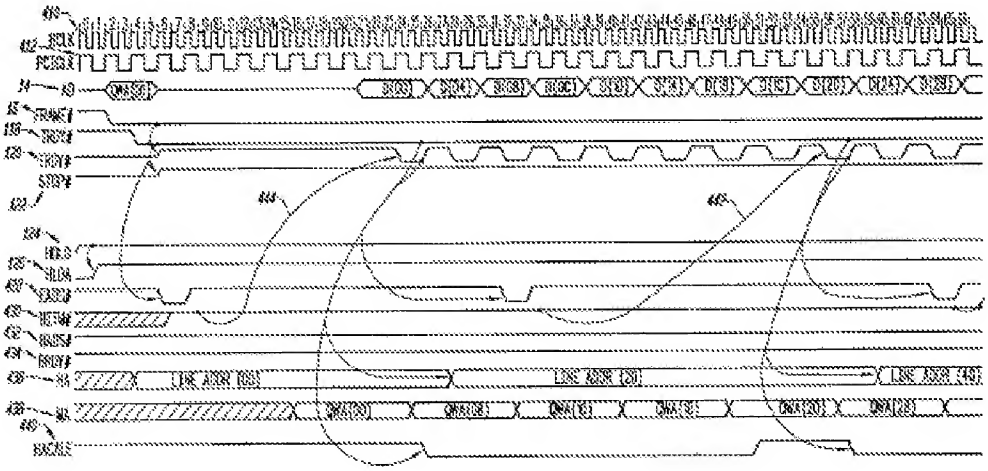


FIG. 4

[EX. 17, '906 Patent, Figure 4].

443. Moreover, this interpretation of Figure 4 is confirmed by the testimony of Subir Ghosh, one of the inventors listed on the '906 patent:

91

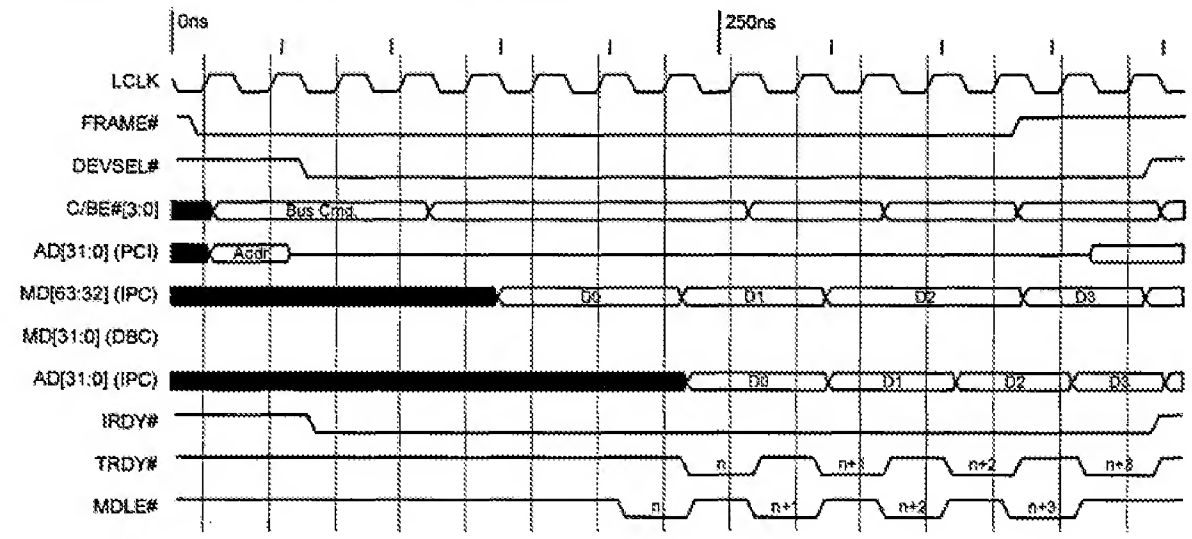
Q. If you could look at Figure 4 again in the patent -- the '906 patent. You don't have to look at the annotated version of it. You can just look at the patent itself.
A. Okay.
Q. In Figure 4, signal line TRDY indicates that one data unit is transferred, then another data unit is transferred, and then another data unit is transferred; is that correct?
A. This one indicates the data units being transferred are always happening with one wait state in between.
Q. The wait state is TRDY nonasserted; correct?
A. That is correct.

444. Documents produced by OPTi are consistent with the '906 patent's description of the OPTi system and Mr. Ghosh's testimony. For example, OPTi produced the

Viper-M 82C556M/82C557M/82C558M Data Book which describes the operation and features of the OPTi Viper-M chipset. [OPTINVIDIA 007132-007389].

445. OPTi's interrogatory responses state that the Viper-M chipset practiced the presnoop claims. The Viper-M Data Book contains Figure 4-26, which purports to illustrate a "PCI Master Read from Local Memory":

Figure 4-26 PCI Master Read from Local Memory

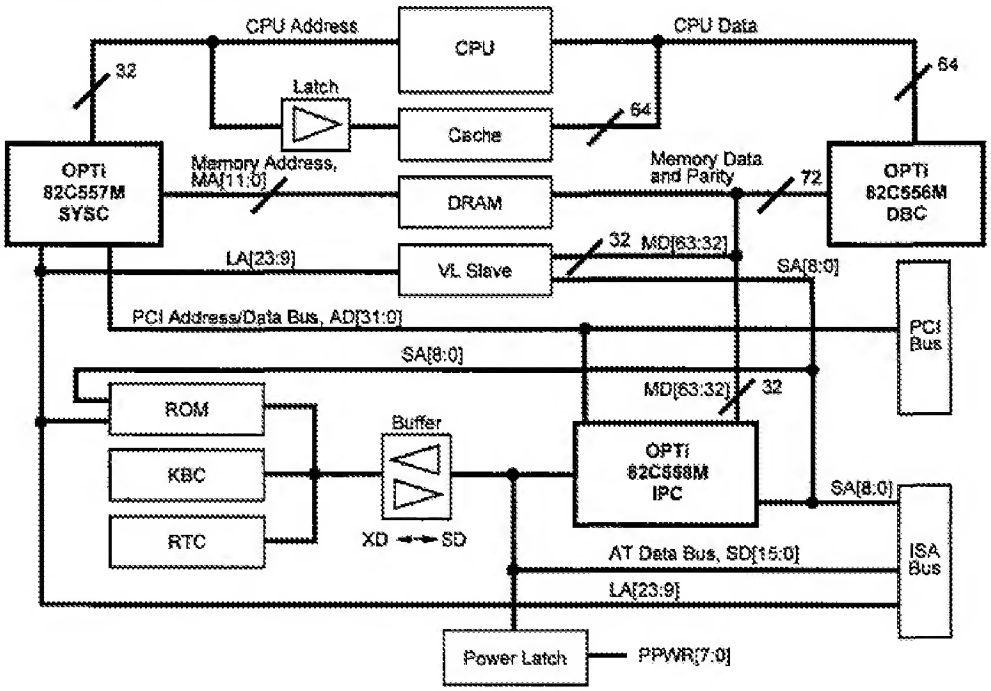


[OPTINVIDIA 007242 (Viper-M Data Book at 89, Figure 4-26)].

446. Figure 4-26 also clearly shows that each 32-bit data transfer on the PCI bus is interrupted by a wait state.

447. The Viper-M Data Book (OPTINVIDIA 007132-007389) suggests why OPTi was unable to achieve the peak data transfer rate available on the PCI bus. In the Viper-M chipset, the PCI local bus was connected to the OPTi 82C558M IPC (Integrated Peripherals Controller) and received all data transfers from main memory (DRAM) via the 82C558M IPC. The system layout is illustrated in Figure 1-1 of the Viper-M Data Book.

Figure 1-1 System Block Diagram



[OPTINVIDIA 007154 (Viper-M Data Book at 1, Figure 1-1)].

448. As Figure 1-1 illustrates, the "PCI Bus" is connected to the "OPTi 82C558M IPC". But as Figure 1-1 also illustrates, the 82C558M IPC is only connected to the upper 32 bits of the MD bus (MD[63:32]). Thus, for a low order DRAM read, the 82C558M needed the assistance of the 82C556M to capture the lower 32-bits of the DRAM read, and drive it back onto the upper half of the MD bus where the 82C558M could latch it and forward it onto the PCI bus:

> For a low order DRAM read, the DRAM puts out the data on the MD[31:0]# lines. The 82C556M latches the data, inverts it and puts it out onto the MD[63:32] lines and drives it out to the 82C558M. The 82C558M then latches the data and puts it out on the AD[31:0] lines for the PCI master. For a low

[OPTINVIDIA 007240 (Viper-M Data Book at 87)].

449. The 82C556M could redirect the low order DRAM read because it contained a "memory to local bus buffer":

2.1 82C556M (DBC) Data Buffer Controller

The 82C556M DBC performs the task of buffering the CPU to the DRAM memory data path. It also performs parity checking.

- CPU to memory data buffer
- CPU to local bus buffer
- Memory to local bus buffer

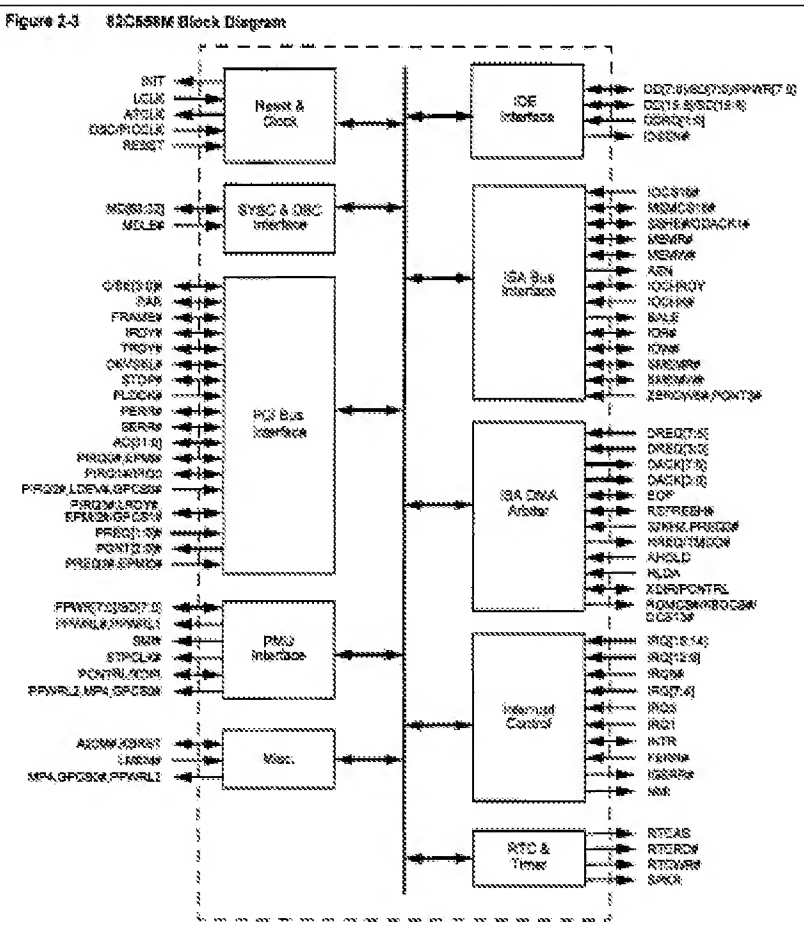
[OPTINVIDIA 007156 (Viper-M Data Book at 3)].

450. The 82C558M could handle high order DRAM transfers by itself because it was connected directly to the high order DRAM signal lines:

> For a high order DRAM read, the DRAM puts out the data on the MD[63:32] lines. In this case, there is a direct path from the DRAM to the 82C558M and the 82C556M does not have to perform any latching or steering of data. The 82C558M latches the data available on the MD[63:32] lines and puts it out on the AD[31:0] lines for the PCI master. For a high order

[OPTINVIDIA 007240 (Viper-M Data Book at 87)].

451. However, the 82C558M could only receive 32-bits of data from the DRAM at a time (via the MD[63:32] signal lines), and does not appear to have had a data buffer where it could store additional data. [*See* OPTINVIDIA 007156 (Viper-M Data Book at 3) (listing features of 82C558M without buffers)]. The 82C558M block diagram also shows only 32 MD signal lines coming into the 82C558M and further confirms this operation:



[OPTINVIDIA 007158 (Viper-M Data Book at 5, Figure 2-3)].

452. Thus, the Viper-M chipset appears to have required wait states between 32-bit data transfers on the PCI bus simply because it could not both drive out data on the PCI bus and simultaneously receive data from the DRAM or 82C556M. Instead, the Viper-M chipset was forced to drive out data onto the PCI bus, then wait a clock cycle while the next 32-bit data unit was latched into the 82C558M, then drive out the next data unit and so on.

453. The Viper-M chipset's inability to transfer 32-bit data units on every PCI clock cycle appears to have severely impacted its performance. As the OPTi patents and Viper-M Data Book indicate, the OPTi device's maximum performance appears to have been only half the peak transfer rate permitted by the PCI Local Bus Specification, or about 66 MB/s. This analysis is consistent with news articles describing the OPTi chipset in 1995.

Opti Fights for Market Share

> Opti's long-awaited Viper-M chip set has been in production since May 1995, about three months behind Triton and Apollo Plus. As Figure 3 shows, Viper-M's system architecture is not as straightforward as that of other chip sets: part of the ISA bus connects to the system controller, and the rest connects to the peripheral controller.
>
> In addition to the common features for a Pentium-class chip set, Viper-m has a few unique features. It supports the Sony Sonic-2WP cache module and adaptive write-back, which means that if a CPU write hits in the secondary cache and is a page hit in main memory, the write will be treated as a write-through cycle. Viper-M can also run its PCI interface in either synchronous or asynchronous modes. In addition, it is the only chip set discussed in this article that supports a VL-bus slave device. The VL support, which complicates the Viper-M architecture, really belongs to 486-class systems. It seems unlikely that this feature will help Opti gain market share.
>
> Opti has been very successful in the chip-set business for many years. Its total market share for desktop PCs was about 20% in 1994. Like VLSI Technology, Opti is losing market share this year for three reasons. First, the Viper-M product has been late to market. Second, the product does not bring enough differentiation in performance and function. In particular, its PCI performance is relatively poor: the sustained data throughout is about 60 Mbytes/s, while all the other chip sets discussed in this article can sustain 100 Mbytes/s or more. Although Opti calls Viper-M a multimedia chip set, it does not have the features to separate it from others as a dedicated multimedia solution. Finally, Intel has simply taken away Opti's market share.
>
> Ironically, Opti has always been thought of as a low-cost chip maker, but it cannot compete with companies like VIA on cost. Opti was once the number-one chip-set supplier to Taiwan, but Opti's presence in the Far East market, which consumes more than 50% of all chip sets worldwide, is not significant any more. Opti seems unable to sustain its profit margins by selling to that market. In the U.S., Intel has stolen major system OEMs like Packard Bell, Gateway, and Dell that used to be either Opti or VLSI customers.

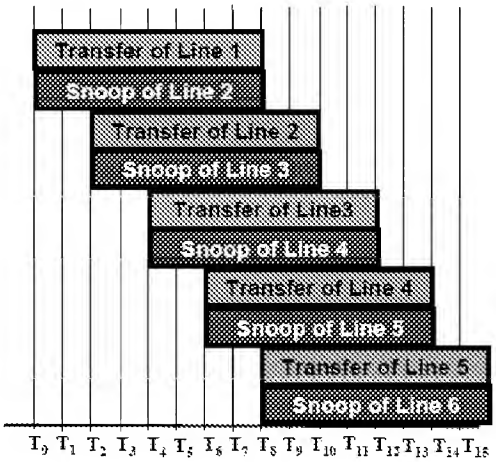
[Yong Yao, Vendors fight for Pentium core-logic market; Intel is king of the land, most others are losing chip-set share (Cover story), MICROPROCESSOR REPORT (August 21, 1995) (NV 021342-021348)].

454. The reason OPTi designed the Viper-M to have such relatively poor PCI performance, and the reason OPTi designed the '906 patent to have such poor performance, is not known, as I understand that OPTi did not retain all of its documents from the relevant time

frame. It is possible that because OPTi was a generation behind Intel and Compaq in PCI chipsets—as Mr. Jaswa testified—that OPTi designed the Viper in a hurried effort to have a PCI chipset on the market, even if it would be much slower than the chipsets that Intel and Compaq were designing. It is also possible that the OPTi designers ran out of gates on the die, or that OPTi had inexperienced designers who didn't realize the impact of their design decisions. In any event, one thing appears clear from the evidence. During the 1994-95 time frame, Intel and Compaq's chipsets were ahead of OPTi in terms of PCI performance.

G. Any claim that covers snooping ahead more than one line is not supported by a written description in the '906 patent specification.

455. OPTi has prepared the following Figure, which OPTi alleges shows how the claimed presnoop method and apparatus works. According to OPTi, the OPTi patent claims cover snooping ahead more than one line.



456. As set forth below, OPTi's Figure is technically incorrect, and conflicts with the teaching of the patent.

1. The Transfer of Line N Cannot Occur Until the Snoop of Line N is Completed.

457. First, the transfer of an individual line cannot take place before the snoop of that line is complete. The whole point of snooping a line is to determine whether or not it is safe to even *begin* accessing the memory locations corresponding to that line – if there is a

snoop hit, that memory transfer cannot start until the dirty cache line has first been written to memory. OPTi's presnoop patents acknowledge that the snoop must be completed before the line transfers. [*See, e.g.*, EX. 17, '906 Patent Col. 5:22-24 (noting that "every data transfer to or from the memory address space which is cached by the L1 cache should be preceded by an inquire cycle")]. Indeed, OPTi's claimed invention is to start the snoop early so that it will complete *before* the earliest possible start of transfer of the line to the PCI agent. Thus, the OPTi figure, which purports to show that the snoop of line 2 is occurring during the transfer of line 2, is incorrect and conflicts with the teaching of the presnoop patents. The snoop of a line quite simply cannot overlap with the transfer of the line, because the snoop must have completed before the transfer of that line can begin.

2. The Snoop of Line N+1 Cannot Occur Until the Snoop of Line N is Completed.

458. Second, OPTi's figure incorrectly shows that more than one line is being snooped at the same time, and between T_6 and T_7, purports to show *four* simultaneous snoops. Simultaneous snoops of different lines are not possible, because each snoop requires exclusive use of the host bus and exclusive use of the snoop port to the host's cache. Only one line may be snooped at a time, and nothing in the patent suggests snooping multiple lines at the same time. No CPU or cache had been invented at the time that allows for more than one line to be snooped at the same time, and nothing in OPTi's patent suggests one exists. Thus, contrary to OPTi's figure, the snoop of line N+1 cannot occur before the snoop of line N is completed.

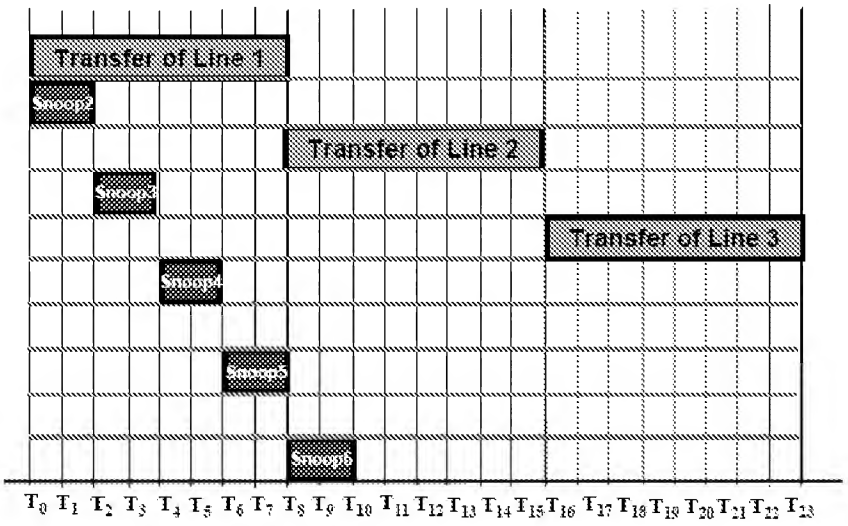
3. The Transfer of Line N+1 Cannot Occur on the Bus Until the Transfer of Line N is Completed.

459. OPTi's Figure shows the transfer of multiple lines on the bus at the same time, where the green "transfer" boxes overlap in the horizontal (time) axis. In fact, only *part* of *one* line may be transferred on a bus at any particular time. More than one line of data cannot be transferred on the PCI bus at a given time – the PCI bus transmits an address for one

cache line, and the data for that line is then transferred to the bus master, one data unit at a time. It is impossible for different lines to be transferred on the PCI bus at a given time. Thus, contrary to OPTi's figure, the transfer of line N+1 on the bus cannot even begin before the transfer of line N is completed, and certainly cannot be concurrent with the transfer of line N.

4. OPTi's Patent Teaches Away from Snooping Ahead More than One Line.

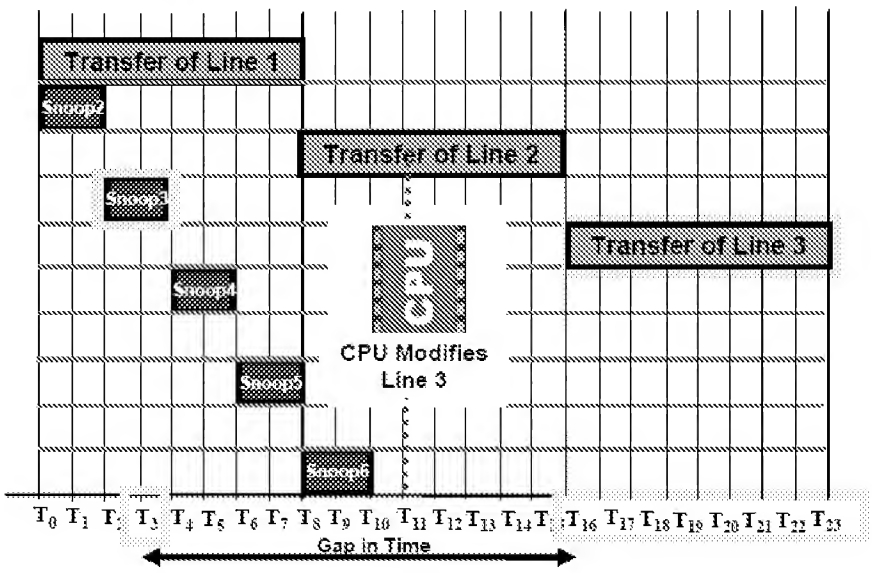
460. In essence, OPTi is attempting to expand its claims to cover snooping ahead more than one line ahead of the line being transferred on the PCI bus, in the manner shown below.



461. The dispute between the parties is whether OPTi's patent claims cover a system that snoops ahead more than the next line – *i.e.,* whether the patent would cover a system where the snoop of line N+2, N+3, or N+4 occurs during the transfer of line N on the PCI bus. In fact, the patent clearly teaches that only the *next line* (N+1) is snooped during the transfer of line N. Snooping ahead more than one line would defeat the purpose of the patent, and the patent teaches away from it. [*See generally* EX. 17, '906 Patent Col. 5:23-25 (noting that extra snoops "would severely hamper the performance of PCI masters performing burst cycles to or from secondary memory.")]. Also note that the timing diagrams of '906 Figures 4-7 show only

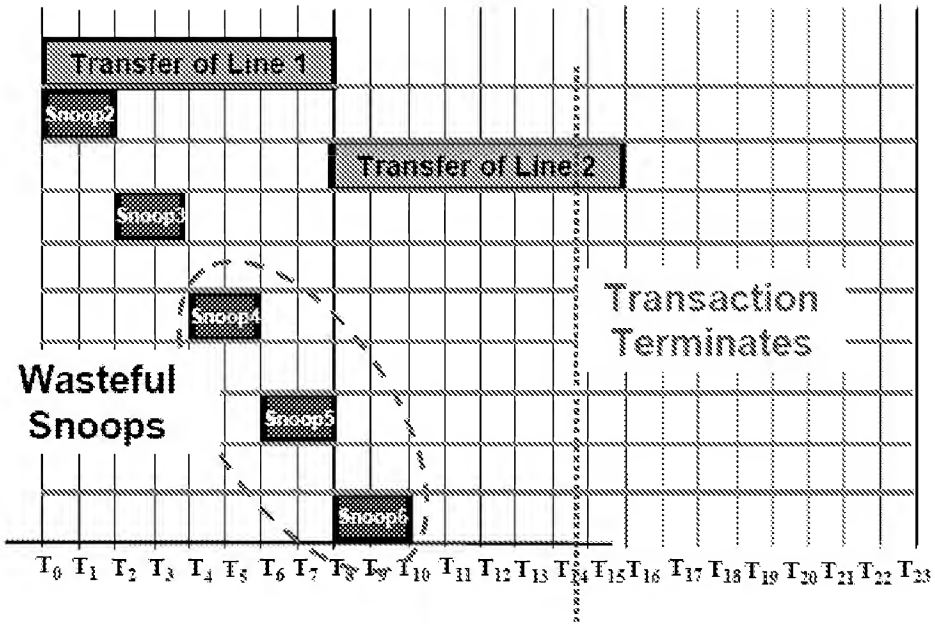
next-line presnooping, and the circuit diagrams are insufficient for any more aggressive forms of presnooping.

462. For example, if line 3 is being transferred many cycles after line 3 is snooped, as OPTi contends, the data in line 3 will be stale if the CPU modified the cache in the meantime. *See* Figure, below.

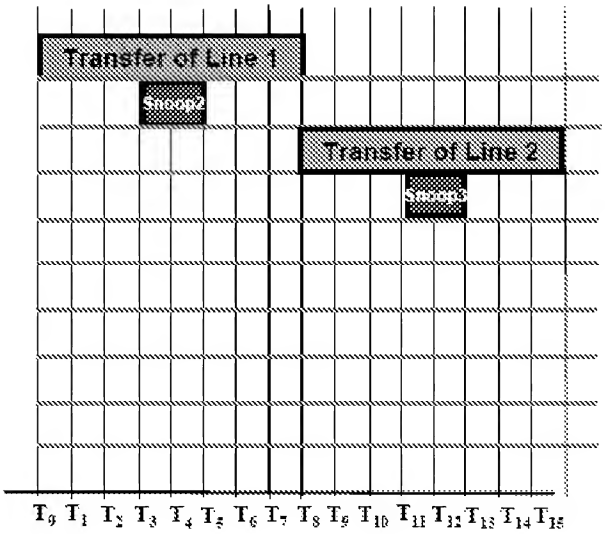


463. The OPTi patent clearly teaches that the "next-line" snoop ensures that the data will not be stale when it is transferred to the bus master. [*See generally* EX. 17, '906 Patent Col. 4:43-46 ("By this process, therefore, the external bus master will be able to access the desired line in main memory *without any further concern that the processor's L1 cache contains a more current copy of the data.*")]. Thus, the OPTi patent clearly teaches away from snooping ahead more than one line, because snooping ahead more than one does not solve the problem, and does not ensure that the bus master receives valid data.

464. Similarly, the OPTi presnoop patents also teach that snoops take time and processor resources, and that unnecessary snoops should be avoided, which is something that persons of skill in the field at the time understood. [*See generally* EX. 17, '906 Patent Col. 5:23-25 (noting that extra snoops "would severely hamper the performance of PCI masters performing burst cycles to or from secondary memory.")]. However, a system that snooped ahead more than one line would by definition perform many senseless snoops, as shown in the Figure below.



465. Where a snoop is initiated more than one line ahead of the line on the PCI bus, multiple senseless snoops would be performed, any or all of which could have instigated associated write backs. Such senseless snoops would burden the processor significantly, and the presnoop patent clearly teaches away from such a system. In fact, the OPTi patent discloses that in every embodiment, the snoop of line N+1 occurs while line N is being transferred on the PCI bus, as set forth at right. For the reasons set forth above, the patent clearly teaches away from any system that snoops more than one line ahead of the line being transferred on the PCI bus.



466. Thus, a system that snoops more than one line ahead of the line on the PCI bus is clearly not described by the OPTi presnoop patents. OPTi argues that persons of skill in the art would have understood that the presnoop invention could be implemented using buffers. Nothing in the patent suggests such a system. The OPTi patents teach away from and discourage any such system, because a system with buffers (1) would have to include another mechanism to make sure that the bus master receives valid data, and (2) would perform senseless snoops. Based on the clear and unambiguous discussion in the presnoop patents, persons of skill in the field would have understood that the presnoop patents teach away from snooping ahead more than one line. The patents describe speculatively snooping ahead only one line ahead of the line being transferred on the PCI bus.

467. I have been provided the following description of the law of written description:

> A patent "must describe the invention sufficiently to convey to a person of skill in the art that the patentee had possession of the claimed invention at the time of the application, i.e., that the patentee invented what is claimed." *LizardTech, Inc. v. Earth Resource Mapping, Inc.*, 424 F.3d 1336, 1345 (Fed. Cir. 2005).
>
> To fulfill the written description requirement, the patent specification "must describe an invention and do so in sufficient detail that one skilled in the art can clearly conclude that the inventor invented the claimed invention." *Lockwood v. American Airlines, Inc.*, 107 F.3d 1565, 1572 (Fed. Cir. 1997). "A patent applicant cannot disclose and claim an invention narrowly and then, in the course of an infringement suit, argue effectively

> that the claims should be construed to cover that which is neither disclosed nor enabled in the patent." *North American Vaccine, Inc. v. American Cyanamid Co.*, 7 F.3d 1571, 1577 (Fed. Cir. 1993). "It is not sufficient for purposes of the written description requirement of §112 that the disclosure, when combined with the knowledge in the art, would lead one to speculate as to modifications that the inventor might have envisioned, but failed to disclose." *Lockwood v. American Airlines, Inc.*, 107 F.3d 1565, 1572 (Fed. Cir. 1997).
>
> Recently, in *LizardTech,* the Federal Circuit affirmed a district court's grant of summary judgment of invalidity for inadequate written description where a patent disclosed "only a single way" of performing the invention and "there is no evidence that the specification contemplates a more generic way." *LizardTech*, 424 F.3d at 1344. The Federal Circuit noted that allowing a patentee to claim a broad, generic invention where the patent failed to describe it "would lead to sweeping, overbroad claims because it would entitle an inventor to a claim scope far greater than what a person of skill in the art would understand the inventor to possess or what a person of skill in the art would be enabled to make and use." *LizardTech*, 424 F.3d at 134

468. The OPTi presnoop patent describe only a single way of predictive snooping – *i.e.,* one line ahead. For the reasons stated above, the patents teach away from anything else. Persons of skill in the art (Intel, Compaq, Motorola) also snooped only one line ahead, for the same reasons. It is my opinion that the OPTi patent does not provide a written description of any invention that snoops more than one line ahead.

XI. CONCLUSION AND RESERVATION.

469. For the reasons set forth above, in the attached claim charts, and in the attached timelines, it is my opinion that clear and convincing evidence shows that the asserted claims of the '906 patent are invalid due to both obviousness and anticipation. Although I have endeavored to provide a comprehensive discussion of my opinions and the basis for them, my analysis and study is ongoing, and I reserve the right to supplement this report or the opinions herein. In particular, I reserve the right to respond to any expert report provided by OPTi, and to address any new or additional information, evidence, or testimony that may be provided to me, including the testimony of fact or expert witnesses at trial and during expert depositions.

Dated: May 16, 2006

Robert P. Colwell

Robert P. Colwell, Ph.D